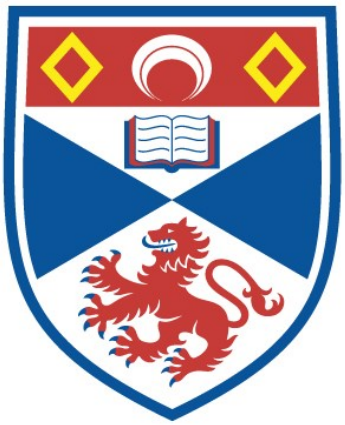
POLYNOMIAL GENERATED POLYGONS

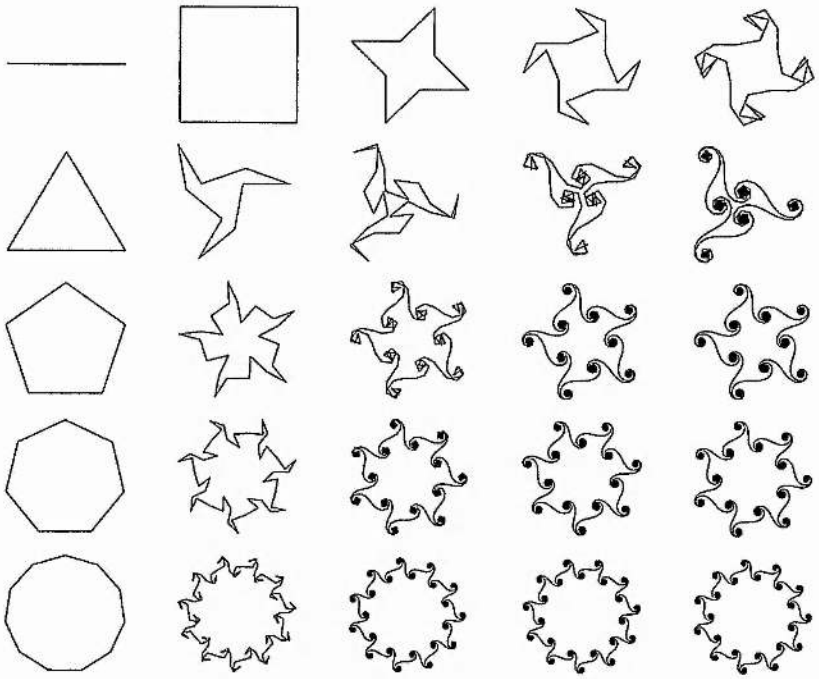
Benedict J. Soares

A Thesis Submitted for the Degree of PhD
at the
University of St Andrews



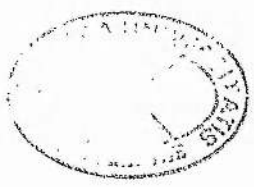
1999

Polynomial Generated Polygons



Benedict J. Soares

SCHOOL OF MATHEMATICAL AND COMPUTATIONAL SCIENCES
UNIVERSITY OF ST ANDREWS
MATHEMATICAL INSTITUTE
NORTH HAUGH
ST ANDREWS
FIFE KY16 9SS
SCOTLAND



ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

TK D406

Abstract

A turtle geometric construction on the plane, called a polynomial generated polygon (PGP) and represented by $\mathcal{P}_{f,p^m}$, is generated from the sequence obtained from evaluating $f(x) \in \mathbb{Z}[x]$ over $\mathbb{Z}$ modulo p^m where p is a prime and $m \in \mathbb{N}$.

Computational methods are developed to pre-calculate the symmetries exhibited by $\mathcal{P}_{f,p^m}$ for a given f and p^m.

These include procedures to find whether $\mathcal{P}_{f,p^m}$ is bounded or unbounded, the degree of rotational symmetry present, whether lines of reflectional symmetry can be observed, and in the case of $\mathcal{P}_{f,p^m}$ unbounded, whether the PGP has a glide reflection.

Methods are also sought to find a suitable f and p^m to produce a desired 'feasible' shape in a PGP construction, and how the same shape might be generated modulo p^{m+k} if it cannot be produced modulo p^m.

Declarations

I, Benedict Soares, hereby certify that this thesis, which is approximately 30000 words in length, has been written by me, that it is the record of work carried out by me and that it has not been submitted in any previous application for a higher degree.

Date: 24/9/99 Signature of candidate: .....

I was admitted as a research student in October, 1994, the higher study for which this is a record was carried out in the University of St Andrews between 1994 and 1998.

Date: 24/9/99 Signature of candidate: .....

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of Ph.D. in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date: 24/9/99 Signature of supervisor: .....

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and abstract will be published, and that a copy of the work may be made and supplied to any *bona fide* library or research worker.

Date: 24/9/99 Signature of candidate: .....

Acknowledgements

My thanks go to John O'Connor for his supervision, help and suggestions;
Edmund Robertson and John O'Connor for the method of construction of a polynomial generated polygon;
and the Caledonian Research Foundation for generous funding during the research for this thesis.

Contents

Part II Classification

Part III Construction

List of Figures

List of Procedures

Part I

Introduction and Preliminaries

Chapter 1

Introduction

1.1 Introduction

We will study certain *turtle geometric shapes* [1]. These are geometrical patterns generated algorithmically from two input variables; a natural number and a polynomial in one variable with integer coefficients. The precise algorithm for generating these shapes is explained in Section 1.2. Examples of the patterns produced appear throughout the the thesis, and in particular in Figures 1.2 on page 8 and 1.1 on page 7 and in Appendix A on page 158.
We attempt in Part II to show how various computational procedures may be used to predict the symmetry group of a shape given the specific polynomial and natural number used in the method of construction.

Many constructions similar to the method examined here are discussed in [1], including a specific example on page 20 is equivalent to our method with the polynomial restricted to a quadratic. Further examples of similar constructions can be found in [6, 8, 7]. [7] is not restricted to a set division of the circle and is more concerned with boundedness than symmetry.

1.2 Polynomial Generated Polygon

A construction on the plane made in a 'natural' way from the sequence generated by an integer-coefficient polynomial evaluated over the integers modulo a given natural number appears likely to display a relatively high degree of symmetry. Often the symmetry group associated with the construction made from such a polynomial, $f(x) \in \mathbb{Z}[x]$, and natural number, $n \in \mathbb{N}$, is that of a cyclic group, C_k, or a dihedral group, D_k, for $k > 2$.

To consolidate the idea of the method used to generate the construction, we present an example followed by the formal definition.

Example

Construction with $f(x) = 3x^2 + x$ and $n = 5$.

Consider a given integer-coefficient polynomial

$$f(x) = 3x^2 + x$$

which we evaluate at $x = 0, 1, 2, \ldots$.
The sequence obtained is

$$(0, 4, 14, 30, 52, 80, 144, 154, 200, 252, \ldots).$$

Now evaluating the terms in this sequence modulo a given natural number, say $n = 5$, gives

$$(0, 4, 4, 0, 2, 0, 4, 4, 0, 2, \ldots).$$

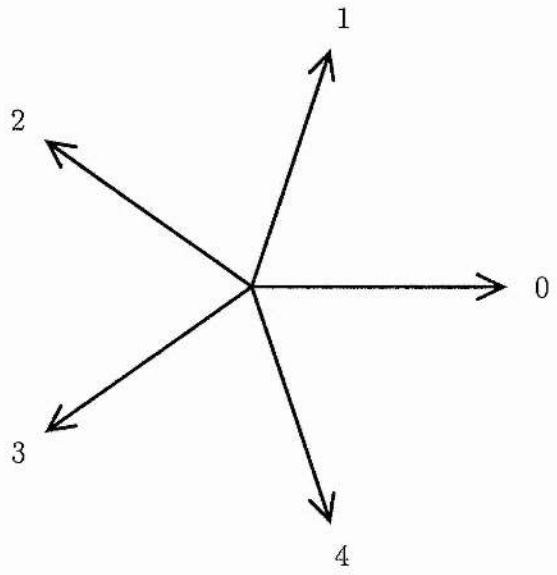
We will construct a geometrical object from this sequence, with an edge of unit length for each entry in the sequence. Each edge will be oriented at an angle associated with the value of that entry, and positioned so that the edge continues from the previous edge (or the origin in the case of the zeroth edge).

Alternatively, we could consider the object as a path on the plane starting at the origin, taking steps of unit length in a direction prescribed by successive terms of the sequence. *i.e.* the j-th step is in the direction associated with $f(j) \bmod n$.

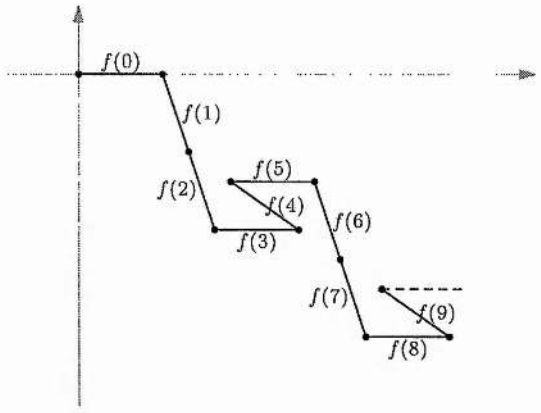
The directions associated with $f(j) \bmod n$ is

$$\frac{2\pi}{n} \times (f(j) \bmod n) \text{ radians from the } Ox\text{-axis.}$$

Going back to our example with $n = 5$, this gives five directions associated with the five possible values of each term in the sequence:

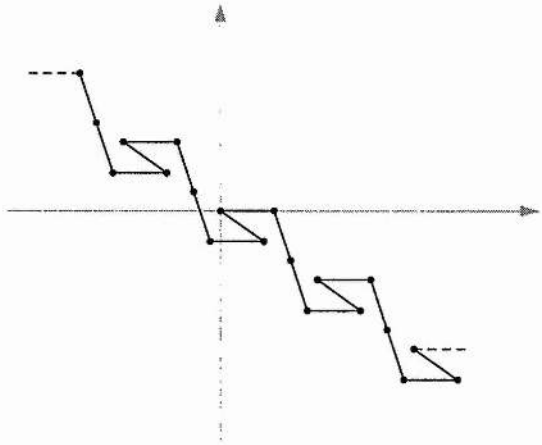


and the construction would look like this:



Since $f(j+n) \equiv f(j) \bmod n$ it is clear that the sequence repeats after n terms. We wish to extend the construction to a bi-infinite sequence $(f(j) \bmod n)$ for $j \in \mathbb{Z}$. We interpret this geometrically by extending the path from the origin in the direction opposite to $(f(j) \bmod n)$ in steps of length 1 for $j = -1, -2, -3, \ldots$. We can thus observe any translational symmetry that may arise.

In our example this results in:



The symmetries of our example construction are generated by a translation in one direction and a rotation of order 2. This is the frieze group $\langle T, R \rangle$ (see Figure 1.3).

We have seen how the construction is made in practice, and now define the construction of a PGP formally.

Definition of a PGP

DEFINITION 1.1: We define the *polynomial generated polygon* (PGP) of an integer-coefficient polynomial f evaluated modulo a natural number n as the construction on the plane that has vertex V_0 at the origin and vertices V_i at the points

$$V_i : \left(\sum_{j=0}^{i-1} \cos\left\{ \frac{2\pi}{n} \times (f(j) \bmod n) \right\}, \sum_{j=0}^{i-1} \sin\left\{ \frac{2\pi}{n} \times (f(j) \bmod n) \right\} \right) \quad (1.1)$$

for $i = 1, 2, \ldots$. We will also consider the PGP to have vertices V_{-i} at

$$V_{-i} : \left(\sum_{j=1}^{i} \cos\left\{ \frac{2\pi}{n} \times (f(-j) \bmod n) + \pi \right\}, \sum_{j=1}^{i} \sin\left\{ \frac{2\pi}{n} \times (f(-j) \bmod n) + \pi \right\} \right) \quad (1.2)$$

for $i = 1, 2, 3, \ldots$. There is an edge (of unit length) from V_i to V_{i+1} is labelled E_i for all $i \in \mathbb{Z}$.

We will denote the PGP generated from $f \in \mathbb{Z}[x]$ and $n \in \mathbb{N}$ by $\mathcal{P}_{f,n}$.

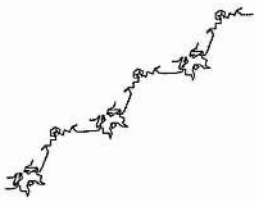
1.3 Further Examples

Many of these constructions lead to complex yet surprisingly symmetrical objects. We present a sample of PGPs in Figures 1.1 on the next page and 1.2. See also Appendix A for examples with $f(x) = ax^3$ with $a = 1, 2, \ldots, n-1$ and $n = 1, 2, \ldots, 29$, and Figure 9.1 for a selection closed PGPs generated using a quadratic f.

Figure 1.1 A selection of values of f and n to generate open PGPs, $\mathcal{P}_{f(x),n}$, displaying various frieze symmetries

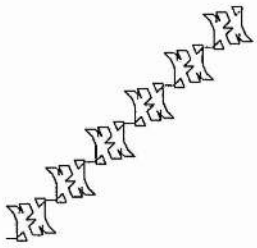
(a) $\mathcal{P}_{x^2+x,125}$

(b) $\mathcal{P}_{2x^2+x,125}$



(c) $\mathcal{P}_{x^3+2x^2,80}$

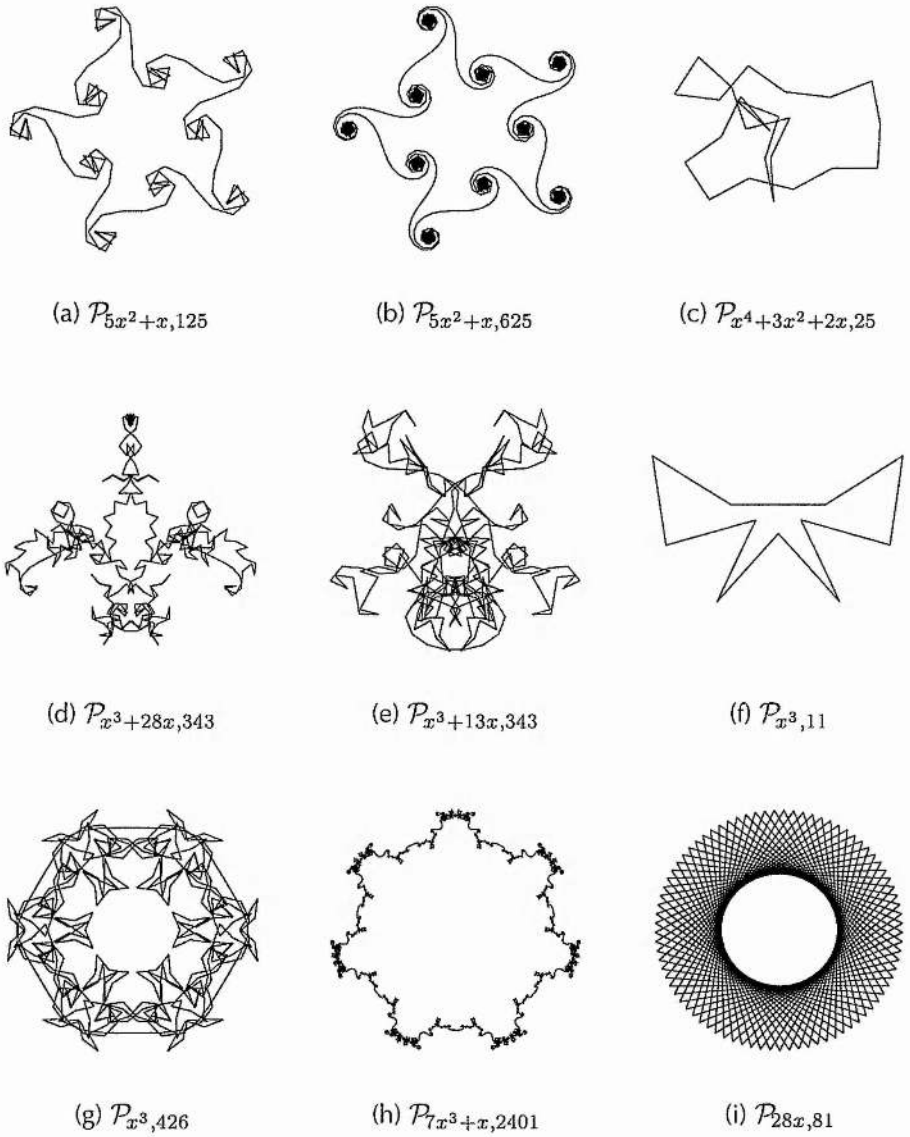
(d) $\mathcal{P}_{x^3,256}$



(e) $\mathcal{P}_{4x^6+3x^4+x^2,64}$

(f) $\mathcal{P}_{4x^6+3x^4+x^2,24}$

Figure 1.2 A selection of values of f and n to generate closed PGPs, $\mathcal{P}_{f(x),n}$, displaying various levels of symmetry



(a) $\mathcal{P}_{5x^2+x,125}$ (b) $\mathcal{P}_{5x^2+x,625}$ (c) $\mathcal{P}_{x^4+3x^2+2x,25}$

(d) $\mathcal{P}_{x^3+28x,343}$ (e) $\mathcal{P}_{x^3+13x,343}$ (f) $\mathcal{P}_{x^3,11}$

(g) $\mathcal{P}_{x^3,426}$ (h) $\mathcal{P}_{7x^3+x,2401}$ (i) $\mathcal{P}_{28x,81}$

1.4 Symmetries

The symmetries of the plane that we expect to find in the constructions can initially be divided into the two categories *bounded* and *unbounded.*

Bounded PGPs

The pattern created from the construction of $\mathcal{P}_{f,n}$ will repeat as the sequence $(f(j) \bmod n)_j$ repeats every n terms. This means that if the n-th vertex is not the same point as the 0-th vertex (the origin), then the translation incurred by this positional discrepancy will be repeated with every n steps of the construction. Consequently the extended construction cannot be bounded.

In terms of bounded constructions, the repetition of the pattern means the n-th vertex must be at the origin, *i.e.* the path joins onto its starting point just as the pattern begins to repeat. For this reason we will refer to bounded objects as *closed.* The symmetries found in these closed objects will be that of the cyclic groups, C_ρ — where ρ-fold rotational symmetry is observed — or the dihedral groups, D_ρ — where ρ-fold rotational symmetry and ρ lines of reflection are found. We prove this assertion in Section 2.5.

Unbounded PGPs

Using the same reasoning as for bounded objects, the constructions will be unbounded precisely if the n-th vertex does not lie exactly on the origin. In this case, translational symmetry will be observed in the extended construction, with a translation in the direction of the n-th vertex from the origin, and a distance that exactly divides the distance from the origin to the n-th vertex.

Since translational symmetry in one direction is always present in an unbounded

construction we will find that the symmetries of such constructions are always described by one of the so-called *frieze groups*.

Each of the seven frieze groups are the symmetry group of patterns that repeat in a translational manner in one direction only, with none or more of the following additional symmetries: a rotation of 180° (R), a reflection in a line parallel to the translation (H), a reflection in a line perpendicular to the translation (V), a glide reflection (a combined translation and reflection in a line parallel to the translation) (G).

Figure 1.3 on the following page shows graphical examples of each of the frieze groups, with the symmetries present, and symmetries suitable for generating the group.

For a rigorous treatment of frieze groups, including a proof that all frieze patterns have one of the seven symmetry groups shown in Figure 1.3, see Chapter 10 of [12].

Questions

We will consider the classification of polynomials, for each n, into their correct symmetry groups. Firstly, given $f \in \mathbb{Z}[x]$ and $n \in \mathbb{N}$, will $\mathcal{P}_{f,n}$ be bounded or unbounded? If bounded, then what is its degree of rotational symmetry? Will it display reflectional symmetry?

If it is unbounded, which of the frieze symmetries (other than translational, which is certain) will be displayed? We endeavour to examine and answer these questions in Part II.

Further questions concern classification of polynomials for each $n \in \mathbb{N}$ by the

Figure 1.3 The seven frieze groups, with their international symbol, symmetries and generators.

p111	T	$\langle T \rangle$
p1a1	T, G	$\langle G \rangle$
pm11	T, V	$\langle T, V \rangle$
p112	T, R	$\langle T, R \rangle$
p1m1	T, G, H	$\langle T, H \rangle = \langle G, H \rangle$
pma2	T, V, G, R	$\langle T, V, R \rangle = \langle G, V \rangle = \langle G, R \rangle$
pmm2	T, R, G, V, H	$\langle T, V, H \rangle = \langle T, R, H \rangle$ $= \langle G, V, H \rangle = \langle G, R, H \rangle$

T is a translation, G is a glide reflection, V is a reflection in a line perpendicular to the translation, H is a reflection in a line parallel to the translation, and R is a rotation of $\frac{\pi}{2}$ [10].

specific shape of $\mathcal{P}_{f,n}$. Which polynomials will produce the same shape? Indeed, given any (feasible) shape, is there a polynomial that will produce that shape for some n? Can we say how many different shapes can be produced given n? These questions will be considered in Part III.

Chapter 2

Preliminaries

2.1 Chinese Remainder Theorem and Prime Powers

One simplification we can make initially is in our choice of values of n. If we know the prime factorisation of n, say

$$n = p_1^{m_1} p_2^{m_2} \dots p_r^{m_r},$$

then we can split some of our questions concerning $\mathcal{P}_{f,n}$ into questions about $\mathcal{P}_{f,p_1^{m_1}}, \mathcal{P}_{f,p_2^{m_2}}, \dots \mathcal{P}_{f,p_r^{m_r}}$.

This is because we can reconstruct $f(x) \bmod n$ from $f(x) \bmod p_i^{m_i}$ for $i = 1, 2, \dots, r$ using the well known *Chinese Remainder Theorem* [15].

Example

Reconstruction of values of $f(x) = 123x^6 + 206x^3 + 12x$ modulo 18 from values calculated modulo 2 and modulo 9.

Given $f(x) = 123x^6 + 206x^3 + 12x$ and evaluating this modulo $18 = 2 \times 3^2$ gives

$$f(x) \equiv 15x^6 + 8x^3 + 12x$$

The sequence $(f(j) \bmod 18)_j$ is

$$(0, 17, 4, 9, 8, 13, 0, 17, 4, 9, 8, 13, 0, 17, 4, 9, 8, 13, \ldots).$$

However, if we had evaluated $(f(j) \bmod 2)$ we would have obtained

$$(0, 1, \ldots)$$

and $(f(j) \bmod 3^2)$ evaluates to

$$(0, 8, 4, 0, 8, 4, 0, 8, 4, \ldots).$$

Now looking at the first entry of $(f(j) \bmod 2)_j$ and $(f(j) \bmod 3^2)_j$ we can see that the first entry of $(f(j) \bmod 18)_j$, a_0, must satisfy both

$$a_0 \equiv 0 \bmod 2$$

and

$$a_0 \equiv 0 \bmod 9.$$

Trivially, 0 satisfies this requirement, and in fact 0 is the only value in $\mathbb{Z}_{18}$ that does, since any such value must be both a multiple of 2 and a multiple of 9, *i.e.* a multiple of $\operatorname{lcm}(2, 9) = 18$. We now move on to the next entry, a_1 that must satisfy

$$a_1 \equiv 1 \bmod 2$$

$$a_1 \equiv 8 \bmod 9.$$

The only value in $\mathbb{Z}_{18}$ to do so is 17. This is less obvious than finding a_0, since now we require a_1 to be a multiple of 2 plus 1 as well as a multiple of 9 plus 8.

i.e.

$$a_1 = \alpha 2 + 1$$

$$a_1 = \beta 9 + 8$$

for some $\alpha, \beta \in \mathbb{Z}$.

Hence

$$9\beta + 7 = 2\alpha \tag{2.1}$$

Looking at (2.1) modulo 9 now tells us that

$$\alpha \equiv 7 \times 2^{-1} \bmod 9$$

where $2^{-1} \bmod 9$ is 5 (2^{-1} modulo 9 is guaranteed to exist because we are using the prime power factors of 18, which are coprime, and so an inverse of one prime power factor modulo a different prime power factor can be found using the well known Euclidean algorithm), and so $\alpha \equiv 35 \equiv 8 \bmod 9$, giving

$$a_1 \equiv 17$$

as required. To check, we see that (2.1) evaluated modulo 2 gives $\beta \equiv -7 \times 9^{-1} \equiv 1 \bmod 2$ yielding the same value for a_1.

We continue in this way and can reconstruct the sequence

$$(f(j) \bmod 18)_j = (0, 17, 4, 9, 8, 13, 0, 17, 4, 9, 8, 13, \ldots)$$

from the sequences $(f(j) \bmod 2)_j$ and $(f(j) \bmod 9)_j$.

PGPs with Prime Powers

The Chinese remainder theorem is a useful procedure because the individual $\mathcal{P}_{f,p_i^{m_i}}$ constructions are simpler to analyse, and some of the properties of $\mathcal{P}_{f,n}$ may be deduced from properties of the $\mathcal{P}_{f,p_i^{m_i}}$.

Some of the properties are as easily deduced for the general case $n \in \mathbb{N}$ as $n = p^m$, however other properties are not easily deduced in this way in the case of a general $n \in \mathbb{N}$, and we must restrict our findings to the case when $n = p^m$.

We will restrict out analysis of the PGPs to those where n is a prime power, p^m, except when we explicitly use n in the construction, when we are referring to any $n \in \mathbb{N}$.

Set of Prime Powers

Since we will be examining polynomials $f \in \mathbb{Z}[x]$ modulo prime powers, we introduce the set

$$\mathbf{P} = \{(p, m) \in \mathbb{N} \times \mathbb{N} \mid p \text{ is prime}\}. \tag{2.2}$$

However, for simplicity and ease of reading, we will represent the elements of $\mathbf{P}$ as p^m $(= (p, m))$ with an implied knowledge of both p and m.

In particular, the input of an element $p^m \in \mathbf{P}$ to a procedure is shorthand for the input of the two elements $p, m \in \mathbb{N}$ so that the procedure may make use of p and m separately without having to factorise $n = p^m$ first.

2.2 Period of Polynomial Sequence — First Repetition

As previously mentioned, the sequence $A = (f(j) \bmod p^m)_j$ is at least p^m periodic, by which we mean the smallest period of A is a divisor of p^m. This is because

$f(x+p^m) \equiv f(x) \bmod p^m$ for all $x \in \mathbb{Z}$, which ensures that the period of A divides p^m, but does not necessarily equal p^m. The only divisors of p^m are $p^{m'}$ for $0 \leqslant m' \leqslant m$ limiting the possibilities.

Determining m' requires observance of a property of polynomial rings over a ring with zero-divisors. $\mathbb{Z}_{p^m}$ is such a ring since it has zero divisors $\alpha p^{m'}$ for $0 < m' < m$, $p \nmid \alpha$.

The complication is that for the sequence A to have period $p^{m'}$, we do not require

$$f(x) - f(x+p^{m'}) \equiv 0 \bmod p^m \tag{2.3}$$

in which the coefficients of $f(x)$ are congruent modulo p^m to the respective coefficients of $f(x+p^{m'})$. It is sufficient for us to have

$$f(j) - f(j+p^{m'}) \equiv 0 \bmod p^m \text{ for } j = 0, 1, 2, \ldots, p^{m'} - 1. \tag{2.4}$$

(2.3) is clearly sufficient for (2.4) but not necessary since, for example,

$$2x^2 + 3x \not\equiv x \bmod 4$$

but

$$2j^2 + 3j \equiv j \bmod 4 \text{ for } j = 0, 1, 2, 3.$$

This is because the difference between $2x^2+3x$ and x, namely $2x(x+1)$ evaluates to 0 modulo 4 for all values of $x \in \mathbb{Z}$.

Polynomials with this property will play a large rôle in many of the procedures defined in the following investigations and so we introduce and classify them together here.

2.3 Zero Evaluating Ideals

We define the subset $\mathcal{Z}_{p^m,x}$ of the polynomial ring $\mathbb{Z}[x]$ as

$$\mathcal{Z}_{p^m,x} = \{f(x) \in \mathbb{Z}[x] \quad | \quad f(j) \equiv 0 \bmod p^m \text{ for } j \in \mathbb{Z}\} \tag{2.5}$$

which is equivalent to

$$\mathcal{Z}_{p^m,x} = \{f(x) \in \mathbb{Z}[x] \quad | \quad f(j) \equiv 0 \bmod p^m \text{for} j = 0, 1, 2, \ldots, p^m - 1\}$$

It is easily seen that the set $\mathcal{Z}_{p^m,x}$ is an ideal of $\mathbb{Z}[x]$. *i.e.* Given any $f(x) \in \mathbb{Z}[x]$ and $g(x), h(x) \in \mathcal{Z}_{p^m,x}$, all three polynomials $f(x)g(x)$, $g(x)f(x)$ and $(g(x)+h(x))$ are in $\mathcal{Z}_{p^m,x}$.

We have seen that this ideal is non-trivial and stated that it will play an important part in the procedures to follow. However, we will delay further analysis of $\mathcal{Z}_{p^m,x}$ until Chapter 5 on page 71.

There we will answer questions concerning a *basis* for $\mathcal{Z}_{p^m,x}$ and calculating whether a given $f(x)$ is a member of $\mathcal{Z}_{p^m,x}$ or not.

We will also present a procedure to find a canonical representative $\bar{f} \in \mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$ of any $f \in \mathbb{Z}[x]$.

2.4 Cyclotomic Polynomials

We will be interested in the minimal polynomials (over $\mathbb{Q}[x]$) for the primitive n-th roots of unity, *i.e.* the polynomial with roots $e^{\{\frac{2\pi i}{n}k\}}$ with $0 < k < n$ and $\gcd(n, k) = 1$.

These minimal polynomials are called the *cyclotomic polynomials* and are usually denoted by $\Phi_n(x)$ where $\Phi_n(x)$ is the minimal polynomial for all the primitive n-th roots of unity.

If we represent $e^{\left\{\frac{2\pi i}{n}\right\}}$ by ξ_n, then $\Phi_n(\xi_n{}^k) = 0$ for $0 < k < n$ and $\gcd(n, k) = 1$.

The cyclotomic polynomials are also such that these are their only roots, hence

$$\Phi_n(x) = \prod_{\substack{0<k<n \\ \gcd(k,n)=1}} (x - \xi_n{}^k).$$

The cyclotomic polynomials are in fact in $\mathbb{Z}[x]$ although their definition is as minimal polynomials over $\mathbb{Q}[x]$.

Calculation of cyclotomic polynomials is computationally easy – see [15].

Examples

The first ten cyclotomic polynomials.

$$\Phi_1(x) = x - 1$$

$$\Phi_2(x) = x + 1$$

$$\Phi_3(x) = x^2 + x + 1$$

$$\Phi_4(x) = x^2 + 1$$

$$\Phi_5(x) = x^4 + x^3 + x^2 + x + 1$$

$$\Phi_6(x) = x^2 - x + 1$$

$$\Phi_7(x) = x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$$

$$\Phi_8(x) = x^4 + 1$$

$$\Phi_9(x) = x^6 + x^3 + 1$$

$$\Phi_{10}(x) = x^4 - x^3 + x^2 - x + 1$$

It might appear that the coefficients of the cyclotomic polynomials are 0 or ± 1. However, this is not true; the first counter-example being

$$\begin{aligned}\Phi_{105}(x) = x^{48} + x^{47} + x^{46} - x^{43} - x^{42} \\ - 2x^{41} - x^{40} - x^{39} + x^{36} + x^{35} + x^{34} + x^{33} + x^{32} \\ + x^{31} - x^{28} - x^{26} - x^{24} - x^{22} - x^{20} + x^{17} + x^{16} \\ + x^{15} + x^{14} + x^{13} + x^{12} - x^9 - x^8 \\ - 2x^7 - x^6 - x^5 + x^2 + x + 1\end{aligned}$$

Many other properties are known about the cyclotomic polynomials, including efficient methods of calculation which makes them suitable for using within the procedures that will follow in Part II [15].

2.5 Finite Symmetry Groups

One of the main properties of the PGPs that we will be interested in are their *symmetry groups*. In particular, we show here that the only finite symmetry groups are the cyclic groups C_n and the dihedral groups D_n.

We define a *symmetry* of a planar figure, F, as an isometry of the plane that maps F onto itself. The set of all symmetries of F forms a group under the usual composition of isometries. For an in depth discussion of symmetry groups of the plane, see [3, 16, 12, 14, 10].

Finite Symmetry Groups

The well-known *Classification Theorem for the Isometries on the Plane* states that each non-identity isometry of the plane is exactly one of the following: *translation*, *rotation*, *reflection*, *glide reflection* [12]. These isometries are categorised as *even* or *odd* depending on whether orientation is preserved by the isometry. The even isometries are *translation* and *rotation*, whilst the odd isometries are *reflection* and *glide reflection*.

We now wish to prove *Leonardo's Theorem*, named after Leonardo da Vinci (1452–1519) who is attributed with proving this theorem (from an architectural perspective) [16].

THEOREM 2.1: A finite group of symmetries is either a cyclic group C_n or a dihedral group D_n.

PROOF: Let G be a finite group of symmetries. Then G cannot contain a non-zero translation or a glide reflection (each of which have infinite order). Hence G consists of rotations and reflections.

We first consider the case when G contains no reflections.

If G contains no trivial rotations (*i.e.* G consists solely of the identity) then $G \cong C_1$.

Otherwise G contains a non-identity rotation $R_{(C,\theta)}$ of an angle θ about the point C.

Now if G contains a non-identity rotation $R_{(D,\phi)}$ of an angle ϕ about the point D,

then G must contain $R^{-1}_{(D,\phi)}R^{-1}_{(C,\theta)}R_{(D,\phi)}R_{(C,\theta)}$ which is a translation, unless $C = D$. Hence G contains only rotations about the point C.

So we now have that all elements in G can be written as rotations $R_{(C,\phi)}$ for some $0 \leqslant \phi < 2\pi$. Let $R_{(C,\alpha)}$ be the smallest such rotation, then G also contains $R^{-1}_{(C,\alpha)} = R_{(C,-\alpha)}$.

Now if $R_{(C,\beta)} \in G$ with $\beta > 0$ then we cannot have $0 < \beta - k\alpha < \alpha$ for any $k \in \mathbb{Z}$ which would contradict the minimality of $R_{(C,\alpha)}$. Hence $\beta = k\alpha$ for some $k \in \mathbb{Z}$ for any $R_{(C,\beta)} \in G$ and so any $R \in G$ is such that $R = R^k_{(C,\alpha)}$ for some $k \in \mathbb{Z}$. This means that $G = C_n$ where n is the order of $R_{(C,\alpha)}$ in G, or $n = 2\pi/\alpha$.

We now consider the case when G contains at least one reflection.

It is easy to see that the even isometries (which includes the identity) of G forms a subgroup of G. By the same considerations in the first case, we can see that this subgroup is isomorphic to the cyclic group C_n.

Now we suppose that G has m reflections. Let $S_\ell \in G$ be a reflection in a line ℓ, then the n odd symmetries $R_{(C,\alpha)}S_\ell, R^2_{(C,\alpha)}S_\ell, ldots, R^n_{(C,\alpha)}S_\ell$ are in G which means that $n \leqslant m$. However, the m odd symmetries when multiplied on the right by S_ℓ give m distinct even symmetries, hence $m \leqslant n$.

Hence $m = n$ and $G = \langle R_{(C,\alpha)}, S_\ell \rangle$.

If $n = 1$ then $G = \langle S_\ell \rangle$. If $n > 1$ then $R_{(C,\alpha)}S_\ell$ is a reflection in ℓ which must contain the point A.

Hence the fintite group of symmetries that contains a reflection is isomorphic to the dihedral group D_n.

Combining these cases proves the theorem.

□

Since a closed PGP is a polygon with finite vertices and edges, it follows that the symmetry group of sucha figure must be finite, and so by the Theorem 2.1 must have a symmetry group that is cyclic if no reflections are present, or dihedral if any reflections are present.

2.6 First, Forward and Finite Differences

First Difference Operator

Another important concept that we will use is that of taking first differences of a polynomial. We define the *first difference*, Δf, of a polynomial f to be such that

$$\Delta f(x) = f(x+1) - f(x).$$

We will also use an iterative form of Δ defining $\Delta^1 = \Delta$ and

$$\Delta^{n+1} f = \Delta \Delta^n f \text{ for } n = 1, 2, 3, \ldots$$

so that

$$\Delta^{n+1} f(x) = \Delta^n f(x+1) - \Delta^n f(x) \text{ for } n = 1, 2, 3, \ldots.$$

The iterations beyond Δ^1 are known as *forward differences*.

Finite Difference Operator

We will also be considering a variant of the first difference operator, the *finite difference* operator, defined as

$$\underset{n}{\Delta} f(x) = f(x+n) - f(x),$$

and referred to as the *n-th difference* of f.

Differences of Sequences

We can apply the first and finite difference operators to sequences of integers in an obvious way.

i.e. if $A = (a_i)_{i=0}^{\infty}$, then the first difference of A, is

$$\Delta A = (a_{i+1} - a_i)_{i=0}^{\infty},$$

and the n-th difference of A is

$$\underset{n}{\Delta} A = (a_{i+n} - a_i)_{i=0}^{\infty}.$$

Differences evaluated modulo m

When applying first or n-th differences to a polynomial f or sequence A, we will often wish to re-evaluate each term modulo m. We will represent these operators

as

$$\Delta_{(m)} f(x) = f(x+1) - f(x) \bmod m$$

$$\Delta_{(m)} A = (a_{i+1} - a_i \bmod m)_{i=0}^{\infty}$$

$$\underset{n}{\Delta}_{(m)} f(x) = f(x+n) - f(x) \bmod m$$

$$\underset{n}{\Delta}_{(m)} A = (a_{i+n} - a_i \bmod m)_{i=0}^{\infty}$$

2.7 Notation for Procedures

We will be formulating many procedures throughout Parts II and III and so a description of the notation that will be used is given here.

Indentation of Blocks

The procedures will be written in a 'pseudo-code' which will hopefully make the structure of each procedure clear and keep the actual function of each step in the procedure unambiguous though easy to follow.
'Blocks' of code ('loops' and 'conditionals') will be indented in the usual way, with a nested block being indented more than its ambient block.
When a return statement is met for the first time, the procedure terminates returning the argument of the return statement. This may happen inside a loop, in which case the loop is terminated at that point.

Assignment and Equality

One of the possible confusions to arise in procedural code is the different notation required for *assignment* and *equality*.

Assignment is where the statement

$a = 2$

means that the variable a is assigned the value 2, whereas *equality* (usually as part of a conditional test) is where the equation

$a = 2$

is evaluated and returns the boolean TRUE if the variable a represents the value 2 at that point in the code, and returns the boolean FALSE otherwise.
In some programming languages this ambiguity is resolved by using the '=' sign to mean assignment and '==' to mean testing of equality. In some other languages assignment is represented by ':=' and equality by '='.
We will be using the notation ':= ' for assignment and '=' for equality.

Name, Domain and Range

In some ways a procedure can be viewed in a similar light to a function, in that it takes an input value (or argument) from a particular domain and outputs a value (or return value) in a particular range. *e.g.* a procedure that calculates the sum of the first n integer values (*i.e.* from 0 to $n-1$) of a given real-coefficient polynomial would take an argument from the domain $\mathbb{R}[x] \times \mathbb{N}$ and would output a return value in the range $\mathbb{R}$. We might regard the procedure as taking two arguments; a value from $\mathbb{R}[x]$ and a value from $\mathbb{N}$. Functions are usually expressed with a name of a single letter (*e.g.* f in $f(x) = x^2$), whereas procedures tend to have names

more decriptive of what they actually do.

We will represent a procedures name, domain and range on the first line of its code, preceded by the word procedure, thus:

procedure **name**($arg_1 \in \text{domain}_1, arg_2 \in \text{domain}_2, \ldots$) $\rightarrow$ (range)

Example

The procedure sum_polynomial.

As an example of this pseudocode, we will present in Procedure 2.1 some code to perform the task mentioned above of calculating the sum of the first n integer values of a given real-coefficient polynomial.

Procedure 2.1 Calculates the sum $\sum_{i=0}^{n-1} f(i)$

procedure **sum_polynomial** ($f \in \mathbb{R}[x], n \in \mathbb{N}$) $\rightarrow$ ($\mathbb{R}$)

$s := 0$

for $i := 0$ to $n - 1$

$s := s + f(i)$

end for

return s

Part II

Classification

Chapter 3

Symmetries

3.1 Introduction

As discussed in Section 1.4 the PGP constructed from a given f and p^m repeats at the p^m-th step, possibly having repeated already at a $p^{m'}$-th step. We prove this simple lemma here, whilst stating more precisely what we mean by 'repeats':

LEMMA 3.1: The pattern observed in $\mathcal{P}_{f,p^m}$ repeats after p^m steps.

PROOF: Given $f(x) \in \mathbb{Z}[x]$, say

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_r x^r,$$

then

$$f(j + p^m) \equiv f(j) \bmod p^m$$

since

$$a_0 + a_1(j + p^m) + a_2(j + p^m)^2 + \ldots + a_r(j + p^m)^r \equiv a_0 + a_1 j + a_2 j^2 + \ldots + a_r j^r \bmod p^m.$$

Hence the sequence used to generate the directions of consecutive edges in $\mathcal{P}_{f,p^m}$ is periodic with a period that divides p^m, *i.e.* $p^{m'}$ for some $0 \leqslant m' \leqslant m$.
It is the repetition of this sequence that implies a repetition in the direction of the edges of $\mathcal{P}_{f,p^m}$ and defines precisely what we mean by $\mathcal{P}_{f,p^m}$ 'repeating'.

□

We have shown that the pattern produced by $\mathcal{P}_{f,p^m}$ repeats at each p^m-th step, which in turn implies that the minimum period of this repetition is $p^{m'}$ for $0 \leqslant m' \leqslant m$. In fact, knowledge of m' is very useful, as many of the procedures in Part II will require this value. How do we calculate m'?

3.2 First Repetition

Finding m' is the same as finding the smallest period of $A = (f(j) \bmod p^m)_j$. We need to check the validity of the set of equations (2.4) for values of $m' = 0, 1, 2, \ldots$ until (2.4) holds. (2.4) will certainly hold for $m' = m$ so this procedure will terminate.

We define Procedure 3.1 to do just this.

Procedure 3.1 Calculates the minimum period of $(f(j) \bmod p^m)_j$

```
procedure first_rep (f ∈ ℤ[x], p^m ∈ P) → (P)
    for m' := 0 to m
        if f(x + p^m') − f(x) ∈ Z_{p^m,x} then
            return p^m'
        end if
    end for
```

It is not clear how the test in line 3 of Procedure 3.1,

$$f(x+p^{m'})-f(x)\in\mathcal{Z}_{p^m,x}, \tag{3.1}$$

is performed. We can replace the test with a call to a procedure

$$\textbf{in\_Z}(f(x+p^{m'})-f(x),p^m)$$

(which will return a boolean value of true or false) to be discussed in Chapter 5 (Procedure 5.3 on page 99).

Notice that if we are given $f\in\mathbb{Z}[x]$ and $n\in\mathbb{N}$, then we can calculate the first repetition, r, of the construction by using the prime factorisation of n.

If

$$n=p_1^{m_1}p_2^{m_2}\dots p_k^{m_k} \tag{3.2}$$

and letting

$$\begin{aligned} A &= (a_{0,j})_j = (f(j))_j \\ A_n &= (a_{n,j})_j = (f(j) \bmod n)_j \\ A_i &= (a_{i,j})_j = (f(j) \bmod p_i^{m_i})_j \text{ for } i=1,2,\dots,k, \end{aligned} \tag{3.3}$$

then $A_i=(a_{i,j})_j=(a_{n,j} \bmod p_i^{m_i})_j=(a_{0,j} \bmod p_i^{m_i})_j$ and so from equations (3.3) we can reconstruct A_n from the A_is using the Chinese remainder theorem.

Also from (3.3) it is clear that the first repetition of A_n, say r_n, must be a multiple of the first repetition of each of the A_is, say r_i for $i=1,2,\dots,k$.

In fact r_n is $l=\operatorname{lcm}(r_1,r_2,\dots,r_k)$ since $a_{i,(l+j)}=a_{i,l}$ for each $i=1,2,\dots,k$ and for all $j\in\mathbb{Z}$ which means that

$$\left.\begin{aligned} a_{n,(l+j)} &\equiv a_{i,(l+j)} \bmod p_i^{m_i} \quad \text{for } i=1,2,\dots,k \\ &\equiv a_{i,j} \bmod p_i^{m_i} \qquad \text{for } i=1,2,\dots,k \end{aligned}\right\} \Rightarrow a_{n,(l+j)}\equiv a_{n,l} \bmod n \tag{3.4}$$

by the Chinese remainder theorem.

Hence we can calculate the first repetition of the PGP of f and n by calculating the r_i using **firstrep**$(f, p_i^{m_i})$ and then calculating $\text{lcm}((r_i)_{i=1}^{k})$.

Example

The first repetition of $f(x) = x^5 + 2x^4 + 3x^3$ modulo 12 is $r = 6$.

The prime factorisation of 12 is $2^2 \times 3$.

We find the first repeat r_1 of $f(x)$ modulo 2^2:

Our first check is

$$\begin{aligned} f(x+2^0) - f(x) &= f(x+1) - f(x) \\ &= (x+1)^5 + 2(x+1)^4 + 3(x+1)^3 - x^5 - 2x^4 - 3x^3 \\ &= 5x^4 + 18x^3 + 31x^2 + 22x + 6 \\ &\equiv x^4 + 2x^3 + 3x^2 + 2x + 2 \bmod 2^2 \\ &\notin \mathcal{Z}_{2^2,x} \end{aligned}$$

Since this has failed, we now check

$$\begin{aligned} f(x+2^1) - f(x) &= f(x+2) - f(x) \\ &= (x+2)^5 + 2(x+2)^4 + 3(x+2)^3 - x^5 - 2x^4 - 3x^3 \\ &= 10x^4 + 56x^3 + 146x^2 + 180x + 88 \\ &\equiv 2x^4 + 2x^2 \bmod 2^2 \\ &\in \mathcal{Z}_{2^2,x} \end{aligned}$$

Hence $r_1 = 2$.

Now to find r_2 we check

$$\begin{aligned} f(x+3^0) - f(x) &= f(x+1) - f(x) \\ &= 2x^4 + x^2 + x \\ &\equiv x^4 + 2x^3 + 3x^2 + 2x + 2 \bmod 3 \\ &\notin \mathcal{Z}_{3,x} \end{aligned}$$

Which only leaves $r_2 = 3$.

Hence, $r = \operatorname{lcm}(r_1, r_2) = 6$.

$$(f(x) \bmod 12)_{j=0}^{11} = (0, 6, 4, 6, 0, 10, 0, 6, 4, 6, 0, 10)$$

Chapter 4

Boundedness and Bounded Symmetries

Having found the point of first repetition, r, within a PGP we can deduce global properties about the PGP from the part produced by $f(j) \bmod p^m$ with $j = 0, 1, \ldots, r-1$ alone, including the symmetries that will be exhibited.

4.1 Closed PGPs

We stated in Section 1.4 that bounded PGPs were in fact *closed* PGPs in the sense that the part of the construction from the 0-th vertex up to the r-th vertex, where r is the point of first repetition, forms a closed polygon.

Equivalently,

$$V_r = V_0.$$

Following on from V_r (and indeed preceding V_0), are repetitions of the same closed polygon exactly overlaying this first polygon.

We now prove the statement made in Section 1.4.

LEMMA 4.1: $\mathcal{P}_{f,p^m}$ is bounded if and only if $\mathcal{P}_{f,p^m}$ is closed.

PROOF: We established in Lemma 3.1 that the sequence of edges repeats after r steps (where r is given by the procedure **in_Z**), and so if (x, y) are the coordinates of V_r, then the coordinates of V_{kr} are (kx, ky).

If $\mathcal{P}_{f,p^m}$ is bounded, then there is an $M \in \mathbb{R}$ such that the distance from the origin to V_i is less than M for all $i \in \mathbb{Z}$.
However, the distance of the vertex V_{kr} from the origin is $d = k\sqrt{x^2 + y^2}$.

If $(x, y) \neq (0, 0)$ then $\sqrt{x^2 + y^2} > 0$, and d can be made greater than M by choosing any $k > M/\sqrt{x^2 + y^2}$.

Hence, if $\mathcal{P}_{f,p^m}$ is bounded, the coordinates of V_r must be $(0, 0)$, and hence $\mathcal{P}_{f,p^m}$ must be closed.

The converse is clear.

□

Lemma 4.1 means that we can refer to PGPs interchangeably as being bounded or closed, or as unbounded or open.

Closed Constructions

Naturally, one of the first questions we might ask about a given $\mathcal{P}_{f,p^m}$ is whether it is open or closed.
One approach to partly answering this question would be to notice that any open

construction can only have a rotational symmetry of order 1 or 2, since a translational symmetry is always present in such PGPs. This means that if the PGP of f and p^m has rotational symmetry of order 3 or more it must be closed. In fact the rotational symmetry of order 2 in a closed PGP is of a 'different nature' (*i.e.* the map which matches an edge with its image under rotation is different) to that of an open construction with the same 'half-turn' symmetry. This difference can be exploited to calculate the closed or open status of any PGP that has a rotational symmetry of order 2 or more.

However, there are PGPs which are closed with no rotational symmetry, other than the trivial one (*e.g.* $\mathcal{P}_{x^4+3x^2+2x,25}$ in Figure 1.2(c) on page 8), and so a more general approach is needed.

We will explore an approach to calculating the closed or open status of a given PGP in Section 4.2.

PGP construction on the Complex Plane

A more general result can be obtained if we reconsider the construction of a PGP to be made on the complex plane, $\mathbb{C}$. Such a consideration means that we can think of the j-th edge as representing a complex number of modulus 1 and argument $(f(j) \bmod p^m) \times \frac{2\pi}{p^m}$, that is, the j-th edge represents

$$e^{\left\{\frac{2\pi i}{p^m}(f(j) \bmod p^m)\right\}}$$

Hence the position of the j-th vertex is simply the sum of the first j edges, *i.e.*

$$\begin{aligned} V_j &= \sum_{k=0}^{j-1} e^{\left\{\frac{2\pi i}{p^m}(f(k) \bmod p^m)\right\}} \\ &= \sum_{k=0}^{j-1} \xi_{p^m}{}^{(f(k) \bmod p^m)} \end{aligned} \tag{4.1}$$

using the ξ notation introduced in Section 2.4.

Now, for the PGP to be bounded Lemma 4.1 requires

$$V_r = V_0 = O, \tag{4.2}$$

where r is the first repeat and O is the origin.

Using this fact we can now formulate

THEOREM 4.2: The PGP $\mathcal{P}_{f,p^m}$ is closed if and only if

$$\Phi_{p^m}(x) \text{ divides } \sum_{j=0}^{r} x^{(f(j) \bmod p^m)}.$$

where r is the first repeat of $\mathcal{P}_{f,p^m}$ given by Procedure 3.1.

PROOF: Combining equations (4.1) and (4.2), $\mathcal{P}_{f,p^m}$ is closed if and ony if

$$V_r = \sum_{j=0}^{r-1} \xi_{p^m}{}^{(f(j) \bmod p^m)} = 0. \tag{4.3}$$

Now equation (4.3) holds if and only if ξ_{p^m} is a root of the (integer coefficient) polynomial

$$F_{(p^m)}(x) = \sum_{j=0}^{r-1} x^{(f(j) \bmod p^m)}.$$

and since the cyclotomic polynomial $\Phi_{p^m}(x)$ is the minimal polynomial for the root ξ_{p^m}, we must have that

$$\Phi_{p^m}(x) \text{ divides } \sum_{j=0}^{r-1} x^{(f(j) \bmod p^m)}.$$

[15]

Conversely, if $\Phi_{p^m}(x)|F(x)$, then $F(\xi_{p^m}) = 0$ and it follows that $V_r = V_0$.

□

Using Theorem 4.2 we can formulate Procedure 4.1 that takes as arguments $f \in \mathbb{Z}[x]$ and $p^m \in \mathbf{P}$ and returns a boolean value of TRUE if $\mathcal{P}_{f,p^m}$ is closed, and returns FALSE otherwise.

Procedure 4.1 Calculates whether $\mathcal{P}_{f,p^m}$ is closed or open.

```
procedure is_closed (f ∈ ℤ[x], p^m ∈ P) → (BOOLEAN)
    r := first_rep(f, p^m)
    F(x) := Σ_{j=0}^{r-1} x^{(f(j) mod n)}
    if Φ_{p^m}(x) | F(x) then
        return TRUE
    else
        return FALSE
    end if
```

Examples

The PGP $\mathcal{P}_{2x^2+x,8}$ is closed.

Following Theorem 4.2, putting $f(x) = 2x^2 + x$ and $p^m = 8$, the sequence $(f(j) \bmod 8)_{j=0}^{7}$ is

$$(0, 3, 2, 5, 4, 7, 6, 1)$$

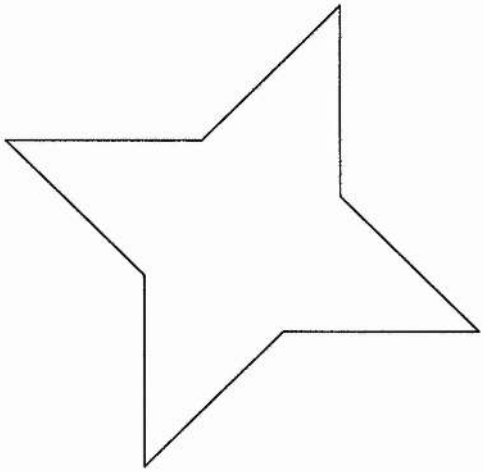
and so

$$\begin{aligned} F(x) &= 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7 \\ &= (x^2 + 1)(x + 1)(x^4 + 1). \end{aligned}$$

Also

$$\Phi_8(x) = x^4 + 1$$

hence $\Phi_8(x)$ divides $F(x)$ and so $\mathcal{P}_{2x^2+x,8}$ is closed.



$\mathcal{P}_{2x^2+x,8}$

The PGP $\mathcal{P}_{x^3+2x,9}$ is closed.

Putting $f(x) = x^3 + 2x$ and $p^m = 9$, the sequence $(f(j) \bmod 9)_{j=0}^{8}$ is

$$(0, 3, 3, 6, 0, 0, 3, 6, 6)$$

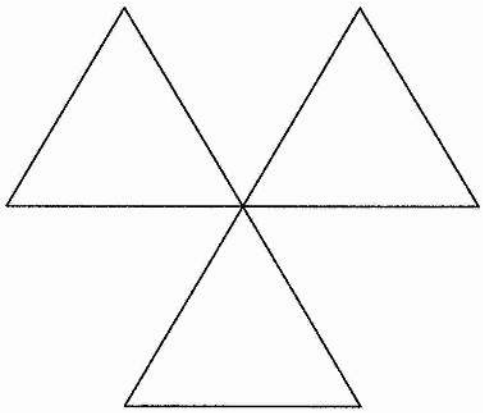
giving

$$\begin{aligned} F(x) &= 3 + 3x^3 + 3x^6 \\ &= 3(x^6 + x^3 + 1). \end{aligned}$$

Also

$$\Phi_9(x) = x^6 + x^3 + 1$$

hence $\Phi_9(x)$ divides $F(x)$ and so $\mathcal{P}_{x^3+2x,9}$ is closed.



$\mathcal{P}_{x^3+2x,9}$

The PGP $\mathcal{P}_{x^4+x^3,16}$ is closed.

Putting $f(x) = x^4 + x^3$ and $p^m = 16$, the sequence $(f(j) \bmod 16)_{j=0}^{15}$ is

$$(0, 2, 8, 12, 0, 14, 8, 8, 0, 10, 8, 4, 0, 6, 8, 0)$$

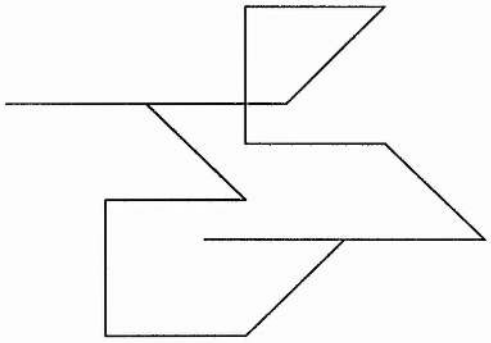
giving

$$\begin{aligned} F(x) &= x^{14} + x^{12} + x^{10} + 5x^8 + x^6 + x^4 + x^2 + 5 \\ &= (x^6 + x^4 + x^2 + 5)(x^8 + 1). \end{aligned}$$

Also

$$\Phi_{16}(x) = x^8 + 1$$

hence $\Phi_{16}(x)$ divides $F(x)$ and so $\mathcal{P}_{x^4+x^3,16}$ is closed.



$\mathcal{P}_{x^4+x^3,16}$

As we can see from the diagram above, this closed PGP has only a trivial rotational symmetry and hence could not be detected as closed using a rotational symmetry method discussed earlier in this section.

The PGP $\mathcal{P}_{x^4+6x^2,8}$ is open.

Putting $f(x) = x^4 + 6x^2$ and $p^m = 8$, the sequence $(f(j) \bmod 8)_{j=0}^{7}$ is

$$(0, 7, 0, 7, 0, 7, 0, 7)$$

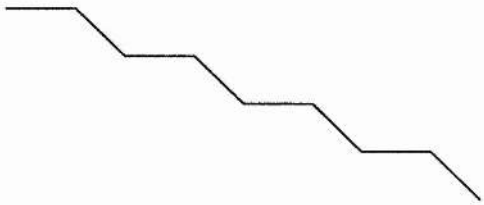
giving

$$\begin{aligned} F(x) &= 4x^7 + 4 \\ &= 4(x+1)(x^6 - x^5 + x^4 - x^3 + x^2 - x + 1). \end{aligned}$$

Whereas

$$\Phi_8(x) = x^4 + 1$$

hence $\Phi_{16}(x)$ does not divide $F(x)$ and so $\mathcal{P}_{x^4+x^3,16}$ is open.



$\mathcal{P}_{x^4+6x^2,8}$

When $m = 1$.

An interesting example of Theorem 4.2 is when $m = 1$, *i.e.* when we are calculating $f(x)$ modulo a prime, p. In such an example, $\Phi_p(x) = 1 + x + x^2 + \ldots + x^{p-1}$. Since the first repeat r is either 1 or p (as it must divide p), we must have $r = p$ for closure to occur. Furthermore, the values of $f(j) \bmod p$ for $j = 0, 1, \ldots, p-1$ must cover the values $0, 1, \ldots, p-1$.

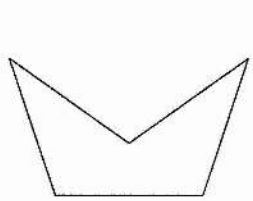
In such a PGP we see all the edges of a regular p-gon, which is clearly a closed curve, though the edges are not necessarily in the usual order. However, the vectorial sum remains zero and so the PGP is obviously closed.

Some examples of such behaviour are

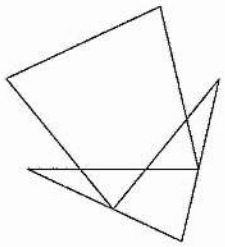
$$\left(j^3 \bmod 5\right)_{j=0}^{4} = (0, 1, 3, 2, 4)$$

$$\left(2j^5 + j^4 + 5j^2 + j \bmod 7\right)_{j=0}^{6} = (0, 2, 4, 6, 1, 5, 3)$$

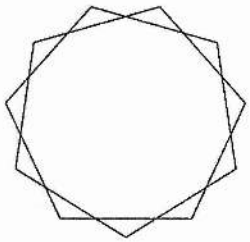
$$\left(2j \bmod 11\right)_{j=0}^{10} = (0, 2, 4, 6, 8, 10, 1, 3, 5, 7, 9)$$



$\mathcal{P}_{x^3,5}$



$\mathcal{P}_{2x^5+x^4+5x^2+x,7}$



$\mathcal{P}_{2x,11}$

Generalising the construction of a PGP Theorem 4.2 leads us to

THEOREM 4.3: For any $F \in \mathbb{Q}[x]$, with say $F(x) = \sum_{j=0}^{r} c_j x^j$, we define

$$F_{(n)}(x) = \sum_{j=0}^{r} c_j x^{(j \bmod n)}.$$

Then

$$\Phi_n(x) \text{ divides } F(x) \text{ if and only if } \Phi_n(x) \text{ divides } F_{(n)}(x).$$

PROOF: We will show two proofs of this theorem, the first using a geometrical argument based on Theorem 4.2, and the second a more direct algebraic argument.

We can imagine a generalised PGP that takes as argument a sequence of integers, $A = (a_i)_{i=0}^{k}$, a sequence of rationals, $L = (l_i)_{i=0}^{k}$ and a natural number, n.

The construction of this generalised PGP, $\mathcal{G}_{A,L,n}$, is similar to that of an ordinary PGP, $\mathcal{P}_{f,n}$ except that instead of placing the i-th edge at an angle

$$\frac{2\pi}{n} \times (f(i) \bmod n) \text{ radians}$$

we place it at an angle

$$\frac{2\pi}{n} \times (a_i \bmod n) \text{ radians}, \tag{4.4}$$

instead of the i-th edge being of unit length, it is now of length l_i, and the construction terminates at the $k+1$-th vertex.

To determine whether $\mathcal{G}_{A,L,n}$ is closed we can use exactly the same arguments as used in the proof of Theorem 4.2, defining the polynomial

$$G_{(n)}(x) = \sum_{i=0}^{k} l_i x^{(a_i \bmod n)}.$$

$\mathcal{G}_{A,L,n}$ is closed if and only if

$$G_{(n)}(\xi_n) = 0$$

$\Leftrightarrow$

$$\Phi_n(x) \text{ divides } G_{(n)}(x). \tag{4.5}$$

Now going back to the construction of $\mathcal{G}_{A,L,n}$, we are evaluating each a_i modulo n, to determine a direction (4.4) to lay the i-th edge. In fact, the evaluation of a_i modulo n is only 'cosmetic', since the direction given by the angle

$$\frac{2\pi}{n} \times (a_i \bmod n) \text{ radians}$$

is, for our purposes, the same as the direction given by the angle

$$\frac{2\pi}{n} \times a_i \text{ radians}.$$

So the construction appears (and is) exactly the same whether we evaluate the terms of A modulo n or not. We will represent $\mathcal{G}_{A,L,n}$ with edge directions *not* evaluated modulo n by $\mathcal{G}^*_{A,L,n}$.

If we define

$$G(x) = \sum_{i=0}^{k} l_i x^{a_i},$$

then $\mathcal{G}^*_{A,L,n}$ is closed if and only if

$$G(\xi_n) = 0$$

$\Leftrightarrow$

$$\Phi_n(x) \text{ divides } G(x). \tag{4.6}$$

However, since $\mathcal{G}^*_{A,L,n}$ is exactly the same as $\mathcal{G}_{A,L,n}$, $\mathcal{G}^*_{A,n}$ is closed if and only if $\mathcal{G}_{A,L,n}$ is closed.

Hence, with (4.5) and (4.6) we have

$$\Phi_n(x) \text{ divides } G(x) \Leftrightarrow \Phi_n(x) \text{ divides } G_{(n)}(x). \tag{4.7}$$

Now, given any $F(x) = \sum_{i=0}^{r} c_i x^i \in \mathbb{Q}[x]$ we can choose k, a_i and l_i for $i = 0, 1, 2, \ldots, k$ thus:

$$k = r$$

$$l_i = c_i$$

and

$$a_i = i,$$

giving

$$G(x) = \sum_{i=0}^{r} c_i x^i = F(x)$$

$$G_{(n)}(x) = \sum_{i=0}^{r} c_i x^{i \bmod n} = F_{(n)}(x).$$

By (4.7) we now have that

$$\Phi_n(x) \text{ divides } F(x) \Leftrightarrow \Phi_n(x) \text{ divides } F_{(n)}(x).$$

This conludes the geometric proof.

Our second proof of this theorem is more direct.

We first note that if and only if $\Phi_n(x)$ divides $F(x)$ and $\Phi_n(x)$ divides $F(x) - F_{(n)}(x)$, then $\Phi_n(x)$ divides $F_{(n)}(x)$.
Hence we need only show

$$\Phi_n(x) \text{ divides } F(x) - F_{(n)}(x).$$

Now

$$\begin{aligned} F(x) - F_{(n)}(x) &= \sum_{j=0}^{r} c_j (x^j - x^{j \bmod n}) \\ &= \sum_{j=0}^{r} \left(c_j x^{j \bmod n} (x^{k_j n} - 1) \right) \quad (k_j \in \mathbb{N} \text{ for all } j). \end{aligned}$$

Now $\Phi_n(x)$ divides $x^n - 1$ which divides $x^{k_j n} - 1$ for all k_j.

This concludes the algebraic proof.

□

Examples

$\Phi_{10}(x)$ divides $x^{12} + x^4 - x^3 - x + 1$

If $F(x) = x^{12} + x^4 - x^3 - x + 1$, then reducing the exponents modulo 10 gives

$$\begin{aligned} F_{(10)}(x) &= x^4 - x^3 + x^2 - x + 1 \\ &= \Phi_{10}(x). \end{aligned}$$

Hence, by Theorem 4.3 $\Phi_{10}(x)|F(x)$.

In fact,

$$\begin{aligned} F(x) &= (x^4 - x^3 + x^2 - x + 1)(x^8 + x^7 - x^3 - x^2 + 1) \\ &= \Phi_{10}(x)(x^8 + x^7 - x^3 - x^2 + 1) \end{aligned}$$

$\Phi_{100}(x)$ divides $x^{122} - x^{112} + x^{42} + x^{40} - x^{32} - x^{30} + x^{20} - x^{10} + x^2 + 1$

If $F(x) = x^{122} - x^{112} + x^{42} + x^{40} - x^{32} - x^{30} + x^{20} - x^{10} + x^2 + 1$, then reducing the exponents modulo 100 gives

$$\begin{aligned} F_{(100)}(x) &= x^{42} + x^{40} - x^{32} - x^{30} + x^{22} + x^{20} - x^{12} - x^{10} + x^2 + 1 \\ &= (x^2 + 1)(x^{40} - x^{30} + x^{20} - x^{10} + 1) \\ &= (x^2 + 1)\Phi_{100}(x). \end{aligned}$$

Hence, by Theorem 4.3 $\Phi_{100}(x)|F(x)$.

In fact,

$$\begin{aligned} F(x) &= (x^{40} - x^{30} + x^{20} - x^{10} + 1)(x^{82} - x^{62} - x^{32} + x^{12} + x^2 + 1) \\ &= \Phi_{100}(x)(x^{82} - x^{62} - x^{32} + x^{12} + x^2 + 1) \end{aligned}$$

$\Phi_7(x)$ does not divide $2x^{10} + x^8 + 3x^5 + x^4 + x$

If $F(x) = 2x^{10} + x^8 + 3x^5 + x^4 + x$, then reducing the exponents modulo 7 gives

$$F_{(7)}(x) = 3x^5 + x^4 + 2x^3 + 2x$$

and since $\Phi_7(x) = x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$ is of degree 6, it cannot divide $F_{(7)}(x)$.

In fact,

$$F(x) = x(x+1)(2x^8 - 2x^7 + 3x^6 - 3x^5 + 3x^4 + x^2 - x + 1)$$

We now look at ways of 'detecting' specific symmetries associated with bounded PGPs.

4.2 Rotational Symmetries

Although yielding some of the more complicated looking PGPs, rotational symmetries are perhaps the easiest to detect in $\mathcal{P}_{f,p^m}$ given f and p^m.

We will be evaluating the sequence $(\Delta f \bmod p^m)_j$ and regarding this sequence as the sequence of *angular differences* of $\mathcal{P}_{f,p^m}$, since $(f(j) \bmod p^m)_j$ is the sequence of angles of $\mathcal{P}_{f,p^m}$.

We shall see that the Δ operator can be thought to have a 'straightening' effect on f.

If $\mathcal{P}_{f,p^m}$ has rotational symmetry of degree ρ, say, then the edges of the PGP coincide when it is rotated by $\frac{360}{\rho}^\circ$ (about its centre), and more importantly, the angular differences also must coincide. Any such rotation corresponds to starting the sequence of angles at another point, which does affect the sequence of the values of the angles, but it does not affect the sequence of the values of the angular differences.

THEOREM 4.4: The PGP $\mathcal{P}_{f(x),p^m}$ has rotational symmetry of degree ρ if and only if the sequence $(\Delta f(j) \bmod p^m)_j$ has a first repeat at r/ρ, where $r = p^{m'}$ $(m' < m)$ is the first repeat of $(f(j) \bmod p^m)_j$ $(\Rightarrow \rho = p^{m''}$ for some $m'' \leqslant m')$.
i.e.

$$\Delta f(x + \tfrac{r}{\rho}) - \Delta f(x) \in \mathcal{Z}_{p^m,x}. \tag{4.8}$$

for some maximum $\rho = p^{m''}$ with $0 \leqslant m'' \leqslant m'$.

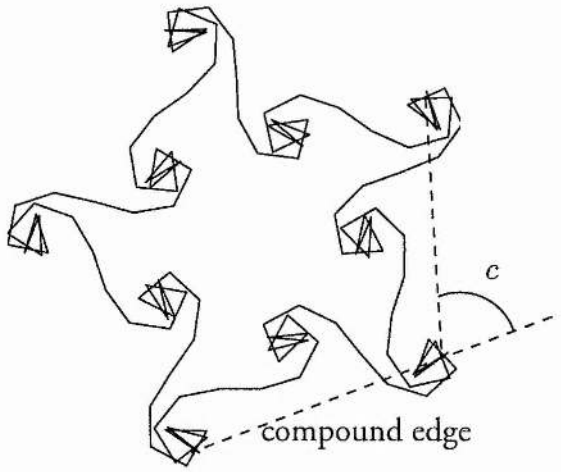
PROOF: We will use the notation $p^m \| c$ with p prime to mean p^m *just* divides c, *i.e.* $p^m | c$ and $p^{m+1} \nmid c$. Given the rotational symmetry of the PGP $\mathcal{P}_{f,p^m}$ is of order ρ and the first repeat of the sequence $A = (f(j) \bmod p^m)_j$ is r, then we have a section of A of length r/ρ being repeated with a constant addition, c say, (with $p^{m-m''} \| c$) to form the segments of $\mathcal{P}_{f,p^m}$ that repeat at a new (constantly differing) angle when the PGP is rotated by $\frac{360c}{p^m}°$. If $\rho = 1$, then c is zero, and $m'' = 0$. If $\rho > 1$, then c is non-zero otherwise we would have a contradictory first repeat at $\frac{r}{\rho} < r$. So we have the value $f(x + \frac{r}{\rho}) \bmod p^m$ is congruent to $f(x) + c \bmod p^m$, where c is non-zero and does not depend on x.
Hence,

$$\underset{r/\rho}{\Delta} f(x) = f(x + \tfrac{r}{\rho}) - f(x) \equiv z(x) + c \bmod p^m \tag{4.9}$$

for some $z \in \mathcal{Z}_{p^m,x}$, and c non-zero with $p^{m-m''} \| c$.

Furthermore, if any $f \in \mathbb{Z}[x]$ satisfies (4.9), then rotational symmetry will be observed.

This is because, given f and p^m such that (4.9) holds, then the first $\frac{r}{\rho}$ edges form the first 'compound edge' of a ρ-gon. The next set of $\frac{r}{\rho}$ edges will repeat the same shaped compound edge, only rotated by a (non-zero) angle of $\frac{360c}{p^m}^\circ$ where c is non-zero. The repetition of ρ of these compound edges results in a polygon with ρ-fold rotational symmetry.



For testing purposes, it is easier to state (4.9) as

$$f(x + \tfrac{r}{\rho}) - f(x) - c \in \mathcal{Z}_{p^m,x}.$$

We will now show that the condition given by equation (4.9) is equivalent to that given by equation (4.8).

Given f such that (4.9) holds,

$$\begin{aligned}
\Delta f(x+\tfrac{r}{\rho}) - \Delta f(x) &= \Big(f(x+\tfrac{r}{\rho}+1) - f(x+\tfrac{r}{\rho})\Big) - (f(x+1) - f(x)) \\
&= \Big(f(x+1+\tfrac{r}{\rho}) - f(x+1)\Big) - \Big(f(x+\tfrac{r}{\rho}) - f(x)\Big) \\
&= \underset{r/\rho}{\Delta} f(x+1) - \underset{r/\rho}{\Delta} f(x) \\
&\equiv (c + z(x+1)) - (c + z(x)) \mod p^m \\
&= z(x+1) - z(x) \\
&\in \mathcal{Z}_{p^m,x}.
\end{aligned}$$

Conversely, if (4.8) holds, then we get

$$\underset{r/\rho}{\Delta} f(x) = c + z(x)$$

for some $z(x) \in \mathcal{Z}_{p^m,x}$.

This c must be non-zero, since it represents the angle turned after the first segment of r/ρ edges. If this were equal to 0 modulo p^m, then the pattern created in this first segment would be repeated without any rotation implying that the first repetition of $(f(j) \bmod p^m)_j$ is $\frac{r}{\rho}$, contradicting the fact that r is the first repetition of $(f(j) \bmod p^m)_j$.

Hence the theorem.

□

Using Theorem 4.4 we can formulate **rot_symm** seen in Procedure 4.2.

Procedure 4.2 takes f and p^m and uses our criteria to test for rotational symmetry of degree $p^{m'-i}$ with $m'-i$ running from m' to 1 where $r = p^{m'}$ is the first repeat

Procedure 4.2 Calculates the degree of rotational symmetry of the closed PGP $\mathcal{P}_{f,p^m}$

```
procedure rot_symm(f ∈ ℤ[x], p^m ∈ P) → (ℕ)
    p^m' := first_rep(f, p^m)
    for i := 0 to m' − 1
        c := f(p^i) − f(0) mod p^m
        if in_Z(f(x + p^i) − f(x) − c, p^m) then
            return p^(m'−i)
        end if
    end for
    return 1
```

discovered by **first_rep.**

If equation (4.9) is satisfied with our prospective rotational symmetry of degree $p^{m'-i}$, then the procedure terminates returning this value.

We should note that the procedure will test the highest possible values of ρ first, as $p^{m'-i}$ decreases.

If no non-trivial rotational symmetry is found, then the procedure will return the value 1, which would be returned if i were to run from 0 to m' in the for loop. However, the final test (if reached) that $f(x+p^{m'}) - f(x) - c \in \mathcal{Z}_{p^m,x}$ has already been performed in obtaining the value of m' earlier.

Examples

The PGP $\mathcal{P}_{3x^2+x,9}$ has rotational symmetry of degree $\rho = 3$.

Given $f(x) = 3x^2 + x$ and $p^m = 9$, the procedure **first_rep** calculates that the first

repetition of $f(j)$ modulo 9 to be $r = p^{m'} = 3^2$.

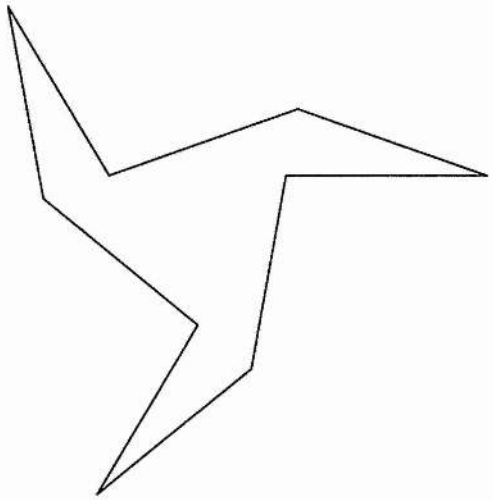
Now putting $i = 0$ we find $c = f(1) - f(0) = 4$, and

$$\begin{aligned}\underset{1}{\Delta} f(x) - c &= f(x+1) - f(x) - c \\ &= 3x^2 + 7x + 4 - 3x^2 - x - 4 \\ &= 6x \\ &\notin \mathcal{Z}_{9,x}.\end{aligned}$$

Now we try with $i = 1$, finding $c = f(3) - f(0) = 30 \equiv 3 \bmod 9$ and

$$\begin{aligned}\underset{3}{\Delta} f(x) - c &= f(x+3) - f(x) - c \\ &= 3x^2 + 19x + 30 - 3x^2 - x - 3 \\ &= 18x \\ &\equiv 0 \bmod 9 \\ &\in \mathcal{Z}_{9,x}.\end{aligned}$$

Hence the degree of rotational symmetry of $\mathcal{P}_{3x^2+x,9}$ is $\rho = 3^{2-1} = 3$.



$\mathcal{P}_{3x^2+x,9}$

The PGP $\mathcal{P}_{15x^3+3x,25}$ has rotational symmetry of degree $\rho = 5$.

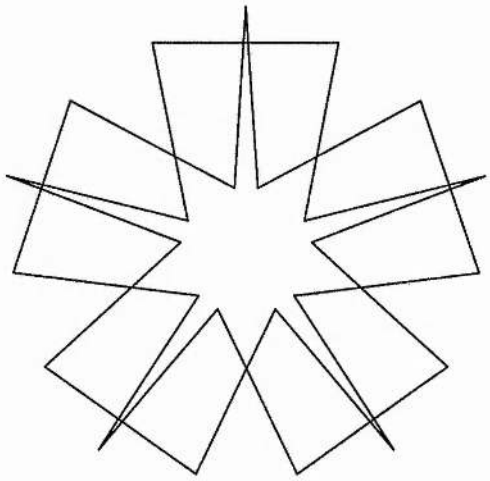
Given $f(x) = 15x^3 + 3x$ and $p^m = 25$, the procedure **first_rep** calculates that the first repetition of $f(j)$ modulo 9 to be $r = p^{m'} = 5^2$.
Now putting $i = 0$ we find $c = f(1) - f(0) = 18$, and

$$\begin{aligned} \underset{1}{\Delta} f(x) - c &= f(x+1) - f(x) - c \\ &\equiv 20x^2 + 20x \bmod 25 \\ &\notin \mathcal{Z}_{25,x}. \end{aligned}$$

Now we try with $i = 1$, finding $c = f(5) - f(0) \equiv 15 \bmod 25$ and

$$\begin{aligned} \underset{5}{\Delta} f(x) - c &= f(x+5) - f(x) - c \\ &\equiv 0 \bmod 25 \\ &\in \mathcal{Z}_{25,x}. \end{aligned}$$

Hence the degree of rotational symmetry of $\mathcal{P}_{15x^3+3x,25}$ is $\rho = 5$.



$\mathcal{P}_{15x^3+3x,25}$

The PGP $\mathcal{P}_{x^4+x^3,16}$ has rotational symmetry of degree $\rho = 1$.

Given $f(x) = x^4 + x^3$ and $p^m = 16$, the procedure **first_rep** calculates that the first repetition of $f(j)$ modulo 16 to be $r = p^{m'} = 2^4$.
Now putting $i = 0$ we find $c = f(1) - f(0) = 2$, and

$$\begin{aligned} \underset{1}{\Delta} f(x) - c &= f(x+1) - f(x) - c \\ &\equiv 4x^3 + 9x^2 + 7x \bmod 16 \\ &\notin \mathcal{Z}_{16,x}. \end{aligned}$$

Now we try with $i = 1$, finding $c = f(2) - f(0) \equiv 8 \bmod 16$ and

$$\begin{aligned}\underset{2}{\Delta} f(x) - c &= f(x+2) - f(x) - c \\ &\equiv 8x^3 + 14x^2 + 12x \bmod 16 \\ &= 2x(4x^2 + 7x + 6) \\ &\notin \mathcal{Z}_{16,x}.\end{aligned}$$

Continuing, $i = 2$, $c = f(4) - f(0) \equiv 0 \bmod 16$ and

$$\begin{aligned}\underset{4}{\Delta} f(x) - c &= f(x+4) - f(x) - c \\ &\equiv 12x^2 \bmod 16 \\ &\notin \mathcal{Z}_{16,x}.\end{aligned}$$

With $i = 3$ we again find $c = f(8) - f(0) \equiv 0 \bmod 16$ and

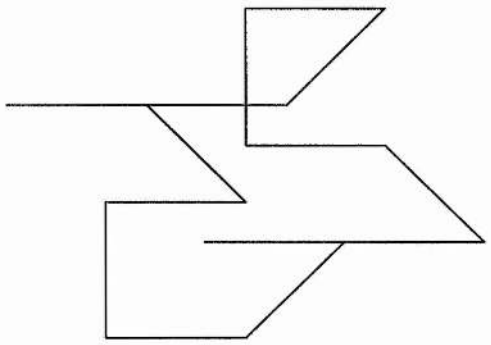
$$\begin{aligned}\underset{8}{\Delta} f(x) - c &= f(x+8) - f(x) - c \\ &\equiv 8x^2 \bmod 16 \\ &\notin \mathcal{Z}_{16,x}.\end{aligned}$$

Now Procedure 4.2 terminates returning the value 1.

Hence the degree of rotational symmetry of $\mathcal{P}_{x^4+x^3,16}$ is $\rho = 1$.

Note that if we were to continue with $i = 4$ we would find

$$\underset{16}{\Delta} f(x) - (f(16) - f(0)) \equiv 0 \bmod 16 \in \mathcal{Z}_{16,x}.$$



$\mathcal{P}_{x^4+x^3,16}$

4.3 Reflectional Symmetries

As seen in Section 2.5, if a line of reflectional symmetry exists in $\mathcal{P}_{f,p^m}$ and $\mathcal{P}_{f(x),p^m}$ has rotational symmetry of degree $\rho = p^{m''}$, then ρ lines of reflective symmetry exist.

If ρ is known, then certainly one of the line(s) of reflective symmetry will intersect $\mathcal{P}_{f(x),p^m}$ in the first $\frac{r}{\rho}$ edges (where $r = p^{m'}$ is the point of first repetition).

A point of intersection must occur either bisecting an angle at a vertex, or bisecting an edge. We will refer to these as an *intersection at a vertex* and an *intersection at an edge* of the line of reflection, R, with $\mathcal{P}_{f(x),p^m}$.

In fact, when p is odd, then each line of reflection must intersect $\mathcal{P}_{f(x),p^m}$ once at a vertex, and once at an edge.

When $p = 2$, then a line of reflection can intersect the PGP at either opposite vertices or opposite edges. Both kinds of line of reflection can occur in the same PGP, however the number of lines of reflection is still equal to the degree of rotational

symmetry, ρ.

To calculate the existence of a line of reflection for a given PGP we require

THEOREM 4.5: The PGP $\mathcal{P}_{f(x),p^m}$ has a line of reflective symmetry intersecting the k-th vertex if and only if

$$\Delta f(k+x) - \Delta f(k-2-x) \in \mathcal{Z}_{x,p^m}. \tag{4.10}$$

$\mathcal{P}_{f(x),p^m}$ has a line of reflective symmetry intersecting the k-th edge if and only if

$$\Delta f(k-1+x) - \Delta f(k-x) \in \mathcal{Z}_{x,p^m}. \tag{4.11}$$

PROOF: We will refer to an angle of $c\frac{360}{p^m}^\circ$ as a p^m-angle of c.

If a line of reflection, R_V, intersects a vertex, V_k, at a p^m-angle c from the Ox axis, then given the bearing of the edge just before this vertex, E_{k-1}, as $f(k-1)$, the direction of the edge following the vertex, E_k must be

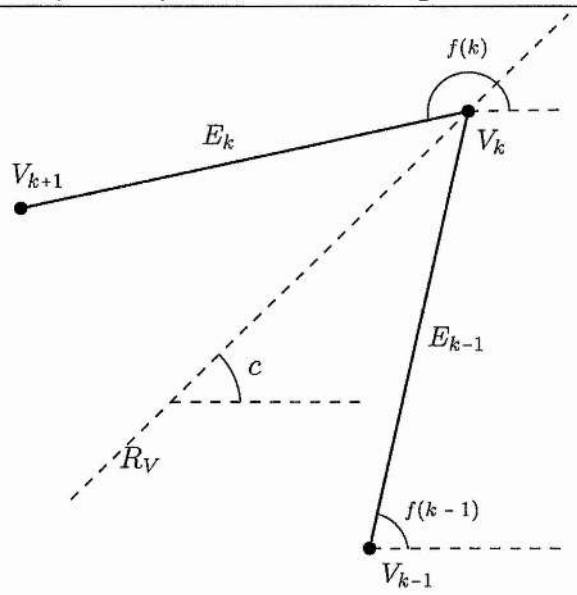
$$f(k) = \tfrac{1}{2}p^m + 2c - f(k-1) \bmod p^m. \tag{4.12}$$

From Figure 4.1 we can see that $c = \frac{1}{2}\left(f(k-1) + f(k)\right) - \frac{1}{4}p^m$.

We can apply the argument used for edges E_k and E_{k-1} to the next 'edges out', E_{k+1} and E_{k-2}, and then subsequent pairs of edges to give

$$f(k+j) = \tfrac{1}{2}p^m + 2c - f(k-j-1) \bmod p^m$$

Figure 4.1 Reflective symmetry about a line through a vertex.



for $j = 0, 1, 2, \ldots$, which implies

$$f(k+x) + f(k-x-1) - f(k-1) - f(k) \in \mathcal{Z}_{x,p^m}. \tag{4.13}$$

If (4.13) holds, then it is clear that the reverse argument is true and that reflective symmetry will be observed around a line bisecting the angle at vertex V_k.

We will now show that (4.13) is an equivalent condition to (4.10).

From (4.10) we have:

$$\Delta f(k+j) - \Delta f(k-2-j) \equiv 0 \bmod p^m$$

i.e.

$$f(k+j+1) - f(k+j) \equiv f(k-1-j) - f(k-2-j) \bmod p^m$$

and so

$$f(k+1+j)+f(k-2-j) \equiv f(k+j)+f(k-1-j) \bmod p^m$$

or

$$f(k+\ell)+f(k-1-\ell) \equiv f(k+(\ell-1))+f(k-1-(\ell-1)) \bmod p^m \quad (4.14)$$

where $\ell = j+1$, for $j = 0, 1, 2, \ldots$.

Applying (4.14) to the RHS of itself j times, yields

$$f(k+\ell)+f(k-1-\ell) \equiv f(k)+f(k-1) \bmod p^m$$

for $\ell = 1, 2, \ldots$ which is equivalent to (4.13).

Hence (4.10) $\Rightarrow$ (4.13).

Now starting from (4.13) we have

$$f(k+j)+f(k-j-1) \equiv f(k-1)+f(k) \bmod p^m$$

for $j = 0, 1, 2, \ldots$.

And so we have

$$f(k+j)+f(k-j-1) \equiv f(k-1)+f(k) \bmod p^m$$

and

$$f(k+j+1)+f(k-j-2) \equiv f(k-1)+f(k) \bmod p^m$$

for $j = 0, 1, 2, \ldots$, and so

$$f(k+j+1)+f(k-j-2) \equiv f(k+j)+f(k-j-1) \bmod p^m$$

or

$$f(k+j+1) - f(k+j) \equiv f(k-j-1) - f(k-j-2) \bmod p^m$$

which implies

$$\Delta f(k+j) \equiv \Delta f(k-j-2) \bmod p^m$$

or

$$\Delta f(k+x) - \Delta f(k-x-2) \in \mathcal{Z}_{p^m,x}.$$

Hence (4.13) $\Rightarrow$ (4.10).

If a line of reflection, R_E, intersects an edge, E_k, at a p^m-angle c from the Ox axis, then given the bearing of the preceding edge, E_{k-1}, as $f(k-1)$, the direction of the succeeding edge, E_{k+1} must be

$$f(k+1) = \tfrac{1}{2}p^m + 2c - f(k-1) \bmod p^m. \tag{4.15}$$

From Figure 4.2 we can calculate $c = f(k) - \frac{1}{4}p^m$.

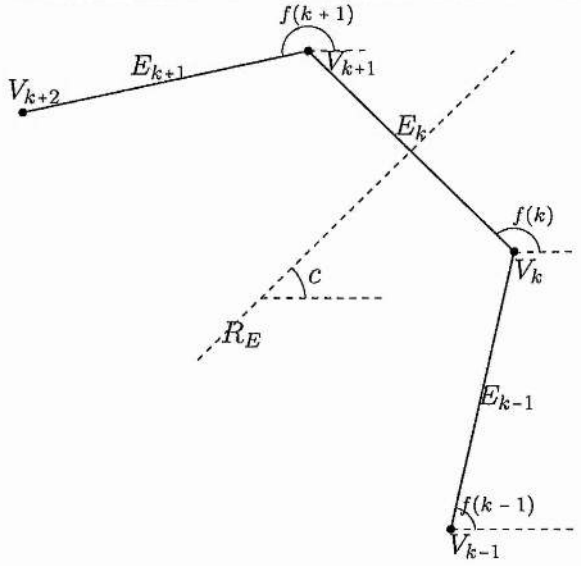
Again, we can apply the argument used for edges E_{k-1} and E_{k+1} to the next edges out, E_{k-2} and E_{k+2}, and then subsequent pairs of edges to give

$$f(k+j) = \tfrac{1}{2}p^m + 2c - f(k-j) \bmod p^m$$

for $j = 0, 1, 2, \ldots$, which implies

$$f(k+x) + f(k-x) - 2f(k) \in \mathcal{Z}_{x,p^m}. \tag{4.16}$$

Figure 4.2 Reflective symmetry about a line through an edge.



If (4.16) holds, then it is clear that the reverse argument is true and that reflective symmetry will be observed around a line bisecting the edge E_k.

We will now show that (4.16) is an equivalent condition to (4.11).

From (4.11) we have:

$$\Delta f(k-1+j) - \Delta f(k-j) \equiv 0 \bmod p^m$$

i.e.

$$f(k+j) - f(k-1+j) \equiv f(k+1-j) - f(k-j) \bmod p^m$$

and so

$$f(k+j) + f(k-j) \equiv f(k+(j-1)) + f(k-(j-1)) \bmod p^m \qquad (4.17)$$

Applying (4.17) to the RHS of itself $j-1$ times, yields

$$f(k+j)+f(k-j) \equiv 2f(k) \bmod p^m$$

for $j = 0, 1, 2, \ldots$ which is equivalent to (4.16).

Hence (4.11) $\Rightarrow$ (4.16).

Now starting from (4.16) we have

$$f(k+j)+f(k-j) \equiv 2f(k) \bmod p^m$$

for $j = 0, 1, 2, \ldots$.

And so we also have

$$f(k+j+1)+f(k-j-1) \equiv 2f(k) \bmod p^m$$

for $j = 0, 1, 2, \ldots$, and so

$$f(k+j+1)+f(k-j-1) \equiv f(k+j)+f(k-j) \bmod p^m$$

or

$$f(k+j+1)-f(k+j) \equiv f(k-j)-f(k-j-1) \bmod p^m$$

which implies

$$\Delta f(k+j) \equiv \Delta f(k-j-1) \bmod p^m$$

Putting $\ell = j+1$ gives

$$\Delta f(k+\ell-1) \equiv \Delta f(k-\ell) \bmod p^m$$

for $\ell = 1, 2, \ldots$, so

$$\Delta f(k-1+x) - \Delta f(k-x) \in \mathcal{Z}_{p^m,x}.$$

Hence (4.16) $\Rightarrow$ (4.11).

This concludes the proof.

□

The geometrical interpretation of equations (4.10) and (4.11) are that of successive angular *differences* around the intersection at a vertex or intersection at an edge are equal. This is similar to the geometrical meaning of the equations (4.13) and (4.16) used in the proof of Theorem 4.5, only the orientation of the PGP doesn't matter; the bearing of the line of reflection isn't needed.

Whether a line of reflection intersects a PGP at a vertex or bisects an edge, the reflection ensures that the angular difference between the edges that meet at the vertices on either side of the intersection are equal. Similarly, the angular differences between the edges that meet at the vertices two away on either side of the intersection are equal.

This helps by limiting the number of vertices and edges that need to be checked for an intersection with a line of reflection in Procedure 4.3 which uses Theorem 4.5 to detect reflective symmetry in a closed PGP generated from a given $f(x)$ and p^m.

Procedure 4.3 Determines whether the PGP $\mathcal{P}_{f(x),p^m}$ has a line of reflection

```
procedure has_ref_symm (f ∈ Z[x], p^m ∈ P) → (BOOLEAN)
    p^m' := first_rep(f, p^m)
    ρ := rot_symm(f, p^m)
    r := p^m' / ρ
    for k := 0 to r − 1
        if in_Z(Δf(k + x) − Δf(k − 2 − x), p^m)
            OR in_Z(Δf(k − 1 + x) − Δf(k − x), p^m) then
            return TRUE
        end if
    end for
    return FALSE
```

Examples

The PGP $\mathcal{P}_{2x^3+2x,9}$ has reflectional symmetry.

Given $f(x) = 2x^3 + 2x$ and $p^m = 9$, the procedure **has_ref_symm** first calculates the first repetition, $p^{m'}$, as 9 using the procedure **first_rep**.

Then the degree of rotational symmetry of $\mathcal{P}_{f(x),p^m}$, ρ, is calculated as 3 using **rot_symm**.

We now need to perform the test formulated in Theorem 4.5 a maximum of $\frac{p^{m'}}{\rho} = 3$ times to determine whether $\mathcal{P}_{f(x),p^m}$ has reflectional symmetry.

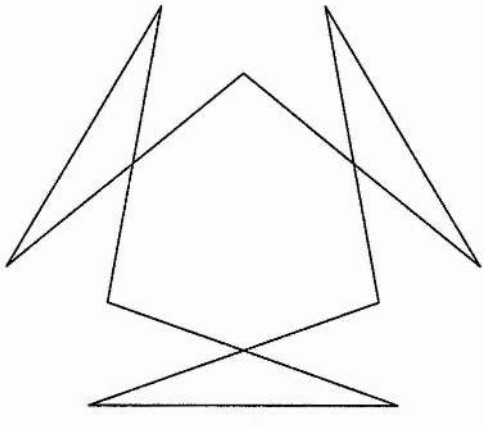
First, with $k = 0$, we calculate

$$\Delta f(k+x) - \Delta f(k-2-x) \equiv 6(x+1) \bmod 9 \notin \mathcal{Z}_{9,x}$$

and

$$\Delta f(k-1+x) - \Delta f(k-x) \equiv 0 \bmod 9 \in \mathcal{Z}_{9,x}$$

and since one of these polynomials is in $\mathcal{Z}_{9,x}$ **has_ref_symm** terminates returning TRUE signifying that $\mathcal{P}_{f(x),p^m}$ has reflectional symmetry.



$\mathcal{P}_{2x^3+2x,9}$

The PGP $\mathcal{P}_{x^5+5x^3+2x,125}$ has (five lines of) reflectional symmetry.

Given $f(x) = x^5 + 5x^3 + 2x$ and $p^m = 125$, the procedure **has_ref_symm** first calculates the first repetition, $p^{m'}$, as 9 using the procedure **first_rep**.

Then the degree of rotational symmetry of $\mathcal{P}_{f(x),p^m}$, ρ, is calculated as 5 using **rot_symm**.

We now need to perform the test found formulated in Theorem 4.5 a maximum of $\frac{p^{m'}}{\rho} = 5$ times to determine whether $\mathcal{P}_{f(x),p^m}$ has reflectional symmetry.

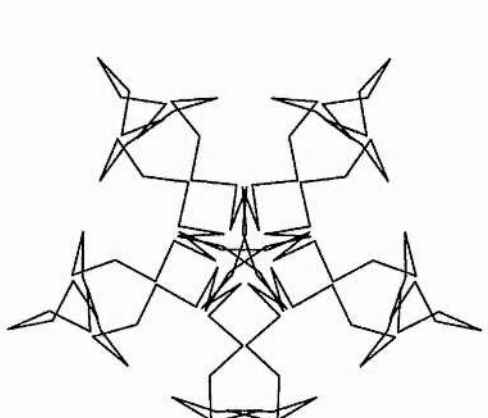
First, with $k = 0$, we calculate

$$\Delta f(k+x) - \Delta f(k-2-x) \equiv 6(x+1) \bmod 9 \notin \mathcal{Z}_{9,x}$$

and

$$\Delta f(k-1+x) - \Delta f(k-x) \equiv 0 \bmod 9 \in \mathcal{Z}_{9,x}$$

and since one of these polynomials is in $\mathcal{Z}_{9,x}$ **has_ref_symm** terminates returning TRUE signifying that $\mathcal{P}_{f(x),p^m}$ has reflectional symmetry.



$\mathcal{P}_{x^5+5x^3+2x,125}$

The PGP $\mathcal{P}_{8x^2+x,128}$ has no lines of reflectional symmetry.

Given $f(x) = 8x^2 + x$ and $p^m = 128$, the procedure **has_ref_symm** first calculates the first repetition, $p^{m'}$, as 128 using the procedure **first_rep**.

The degree of rotational symmetry of $\mathcal{P}_{f(x),p^m}$, ρ, is then calculated as 16 using **rot_symm**.

We now need to perform the test found formulated in Theorem 4.5 a maximum of $\frac{p^{m'}}{\rho} = 8$ times to determine whether $\mathcal{P}_{f(x),p^m}$ has reflectional symmetry.

First, with $k = 0$, we calculate

$$\Delta f(k+x) - \Delta f(k-2-x) \equiv 32(x+1) \bmod 128 \notin \mathcal{Z}_{128,x}$$

and

$$\Delta f(k-1+x) - \Delta f(k-x) \equiv 16(2x-1) \bmod 128 \notin \mathcal{Z}_{128,x}.$$

With $k = 1$, we get

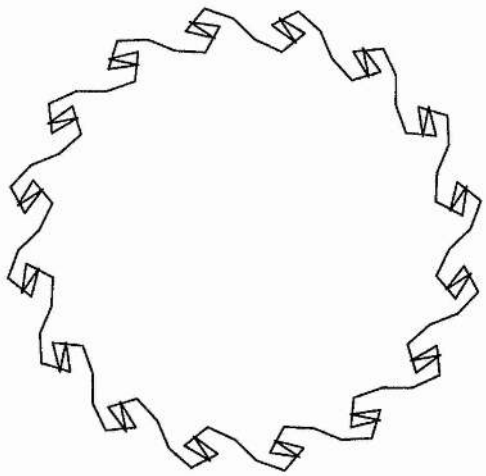
$$\Delta f(k+x) - \Delta f(k-2-x) \equiv 32(x+1) \bmod 128 \notin \mathcal{Z}_{128,x}$$

and

$$\Delta f(k-1+x) - \Delta f(k-x) \equiv 16(2x-1) \bmod 128 \notin \mathcal{Z}_{128,x}.$$

In fact we find the same for $k = 0, 1, 2, \ldots, 7$.

Since none of these polynomials are in $\mathcal{Z}_{128,x}$ **has_ref_symm** completes the loop and then terminates returning **FALSE** signifying that $\mathcal{P}_{f(x),p^m}$ has no reflectional symmetry.



$\mathcal{P}_{8x^2+x,128}$

4.4 Classification of Bounded Symmetries

The symmetry of any bounded PGP must be either a cyclic group, C_ρ, where rotational symmetry of degree ρ is present and reflectional symmetry is not, or a dihedral group, D_ρ, where rotational symmetry of degree ρ is present, and ρ lines of reflection exist.

Our two procedures **rot_symm** and **has_ref_symm** can both be performed to determine the symmetry group of $\mathcal{P}_{f(x),p^m}$ if it is bounded, seen in Procedure 4.4.

Procedure 4.4 Calculates the symmetry group of $\mathcal{P}_{f(x),p^m}$ if it is bounded.

```
procedure bounded_symm (f ∈ ℤ[x], p^m ∈ P) → ( SYMMETRY GROUP )
    ρ := rot_symm(f, p^m)
    if has_ref_symm(f, p^m) then
        return D_ρ
    else
        return C_ρ
    end if
```

However, this slightly simplistic combination of **rot_symm** and **has_ref_symm** is inefficient in that, as seen in Procedure 4.3, **has_ref_symm** also calculates ρ using **rot_symm** and so **rot_symm** is performed *twice* in Procedure 4.4.
We should also note that the procedure **first_rep** is performed in both **rot_symm** and **has_ref_symm** and so will also be performed twice in Procedure 4.4.

We can remedy this by writing a more efficient version of **bounded_symm** that combines **rot_symm** and **has_ref_symm** in a closer way, and only calculates

first_rep$(f(x), p^m)$ and **rot_symm**$(f(x), p^m)$ once. This can be seen in Procedure 4.5.

Procedure 4.5 Calculates the symmetry group of $\mathcal{P}_{f(x),p^m}$ if it is bounded (improvement on Procedure 4.4).

procedure **bounded_symm** $(f \in \mathbb{Z}[x], p^m \in \mathbf{P}) \rightarrow$ (SYMMETRY GROUP)

$p^{m'} :=$ **first_rep**(f, p^m)

$\rho := 1$

for $i := 0$ to $m' - 1$

$c := f(p^i) - f(0) \bmod p^m$

if **in_Z**$(f(x + p^i) - f(x) - c, p^m)$ then

$\rho := p^{m'-i}$

end if

end for

$r := \frac{p^{m'}}{\rho}$

for $k := 0$ to $r - 1$

if **in_Z**$(\Delta f(k + x) - \Delta f(k - 2 - x), p^m)$

OR **in_Z**$(\Delta f(k - 1 + x) - \Delta f(k - x), p^m)$ then

return D_ρ

end if

end for

return C_ρ

Chapter 5

Zero Evaluating Ideals

We repeat here the definition of the zero evaluating ideal of $\mathbb{Z}_{p^m}[x]$ first mentioned in Section 2.3 on page 18.

5.1 Definition

DEFINITION 5.1: The *zero evaluating ideal* of $\mathbb{Z}_{p^m}[x]$, denoted by $\mathcal{Z}_{p^m,x}$, is the set

$$\mathcal{Z}_{p^m,x} = \{f(x) \in \mathbb{Z}_{p^m}[x] \quad | \quad f(j) \equiv 0 \bmod p^m \quad \forall j \in \mathbb{Z}_{p^m}\}.$$

We show that $\mathcal{Z}_{p^m,x}$ is an ideal of $\mathbb{Z}_{p^m}[x]$:

Given $f, g \in \mathcal{Z}_{p^m,x}$ and $h \in \mathbb{Z}_{p^m}[x]$,

$$f(j) + g(j) \equiv 0 + 0 \bmod p^m \text{ for } j = 0, 1, 2, \ldots, p^m - 1$$

and so $f + g \in \mathcal{Z}_{p^m,x}$.

Also,

$$f(j) \cdot h(j) \equiv 0 \cdot h(j) \equiv 0 \bmod p^m \text{ for } j = 0, 1, 2, \ldots, p^m - 1$$

hence $f \cdot h \in \mathcal{Z}_{p^m,x}$.

$\mathcal{Z}_{p^m,x}$ is non-empty since it contains the zero polynomial, 0. It is non-trivial since the monic polynomial $x(x-1)\cdots(x-(p^m-1)) \in \mathcal{Z}_{p^m,x}$.

5.2 Use of Zero Evaluating Ideals

As we have seen in the procedures formulated so far, it is useful to be able to check whether a given polynomial is a member of $\mathcal{Z}_{p^m,x}$.
We should also note that if two polynomials $f, g \in \mathbb{Z}[x]$ differ by a third polynomial $z \in \mathcal{Z}_{p^m,x}$, then the sequence of numbers they produce modulo p^m will be the same. Consequently, the PGPs $\mathcal{P}_{f(x),p^m}$ and $\mathcal{P}_{g(x),p^m}$ will be identical.

Given that $\mathcal{Z}_{p^m,x}$ always contains the monic polynomial $z(x) = x(x-1)\cdots(x-(p^m-1)) \in \mathcal{Z}_{p^m,x}$ we can practically *reduce* a polynomial $f(x)$ to one of degree less than p^m, the degree of z. The resulting polynomial, $\bar{f}$, can be used instead of f when investigating $\mathcal{P}_{f(x),p^m}$. In fact, we will see in this chapter that for $p^m > 4$, $m > 1$ we can always reduce $f(x)$ to a polynomial of degree less than $p^m - 1$. We will also be able to reduce the value of the coefficients of the lower exponents of $f(x)$ to give a fully reduced polynomial $\bar{f}(x)$ that generates the same sequence as $f(x)$ modulo p^m. This polynomial will be a canonical representative of the cosets of $\mathbb{Z}_{p^m}[x]/\mathcal{Z}_{p^m,x}$ and may be thought of as the 'lowest' such element of its coset. By this we mean the polynomial element of a given coset with lowest degree, and with lowest coefficients of each exponent.

Considering the ring homomorphism $\psi : \mathbb{Z}[x] \to (\mathbb{Z}_{p^m})^{p^m}$ defined by

$$f \mapsto (f(0) \bmod p^m, f(1) \bmod p^m, \ldots, f(p^m - 1) \bmod p^m) \tag{5.1}$$

we observe that $\mathcal{Z}_{p^m,x} = \operatorname{Ker} \psi$.

5.3 Bases for Zero Evaluating Ideals

To reduce a given polynomial to its coset representative, we will need to calculate a basis for the zero evaluating ideal $\mathcal{Z}_{p^m,x}$.

LEMMA 5.1: For a given degree, say s, the monic polynomial

$$z(x) = x(x-1)\cdots(x-(s-1)),$$

of degree s, always evaluates to 0 modulo p^t where

$$t = \left\lfloor \frac{s}{p} \right\rfloor + \left\lfloor \frac{s}{p^2} \right\rfloor + \left\lfloor \frac{s}{p^3} \right\rfloor + \ldots. \tag{5.2}$$

PROOF: Here t represents the largest power of p that divides $s!$. The sum on the right hand side of equation (5.2) is a finite sum, since the terms above and including $\left\lfloor \frac{s}{p^\ell} \right\rfloor$ are all zero, where ℓ is the smallest integer greater than $\frac{\log s}{\log p}$.
For any given $j \in \mathbb{Z}$, the value of $z(j)$ modulo p^t is the same as that of $z(j + p^t)$

modulo p^t, hence we may assume that $j > p^t$. Thus,

$$\begin{aligned} z(j) &= j(j-1)\cdots(j-(s-1)) \\ &= \frac{j!}{(j-s)!} \\ &= \binom{j}{s} s! \\ &\equiv 0 \bmod p^t \end{aligned}$$

□

From Lemma 5.1 we deduce that, if $m > t$, the polynomial $p^{m-t}z(x)$ always evaluates to 0 modulo p^m and hence $p^{m-t}z(x) \in \mathcal{Z}_{p^m,x}$.

This means that given any polynomial $f(x) \in \mathbb{Z}[x]$ of degree $s' > s$, we can subtract a polynomial multiple of $p^{m-t}z(x)$ to produce a reduced version of f, say $\bar{f}$, such that the coefficients of $f(x)$ for the powers greater than or equal to s are less than p^{m-t}, and the sequence produced by $\bar{f}(x)$ modulo p^m is the same as that produced by $f(x)$.
Clearly, if $t \geqslant m$, then $\bar{f}$ is of degree less than s.

What we need to find is the smallest s such that

$$m \leqslant \left\lfloor \frac{s}{p} \right\rfloor + \left\lfloor \frac{s}{p^2} \right\rfloor + \left\lfloor \frac{s}{p^3} \right\rfloor + \ldots$$

Calculation of $\mathbf{s}_{p^m}$

THEOREM 5.2: If the smallest $s \in \mathbb{N}$ such that

$$m \leqslant \left\lfloor \frac{s}{p} \right\rfloor + \left\lfloor \frac{s}{p^2} \right\rfloor + \left\lfloor \frac{s}{p^3} \right\rfloor + \dots \tag{5.3}$$

is denoted by $\mathbf{s}_{p^m}$ for a given prime, p, then

$$\mathbf{s}_{p^m} = cp^r + \mathbf{s}_{p^d} \tag{5.4}$$

where

$$\begin{aligned} r &= \left\lfloor \tfrac{\log(mp-m+1)}{\log p} \right\rfloor \\ c &= \left\lfloor \tfrac{m(p-1)}{p^r-1} \right\rfloor \\ d &= m - c\left(\tfrac{p^r-1}{p-1}\right) \end{aligned} \tag{5.5}$$

PROOF: Let us denote the value given on the right-hand-side of the inequality (5.3) by

$$\mathbf{m}_{s,p} = \left\lfloor \frac{s}{p} \right\rfloor + \left\lfloor \frac{s}{p^2} \right\rfloor + \left\lfloor \frac{s}{p^3} \right\rfloor + \dots .$$

$\mathbf{m}_{s,p}$ represents the power of p that divides $s!$. If s were to equal another power of p, say $s = p^r$, then

$$\begin{aligned} \mathbf{m}_{s,p} &= \mathbf{m}_{p^r,p} \\ &= \left\lfloor \frac{p^r}{p} \right\rfloor + \left\lfloor \frac{p^r}{p^2} \right\rfloor + \left\lfloor \frac{p^r}{p^3} \right\rfloor + \dots \\ &= p^{r-1} + p^{r-2} + p^{r-3} + \dots + 1 \\ &= \tfrac{p^r-1}{p-1}. \end{aligned}$$

If $s = ap^r$ $(a < p)$, then

$$\begin{aligned}\mathbf{m}_{s,p} &= \mathbf{m}_{ap^r,p} \\ &= \left\lfloor \frac{ap^r}{p} \right\rfloor + \left\lfloor \frac{ap^r}{p^2} \right\rfloor + \left\lfloor \frac{ap^r}{p^3} \right\rfloor + \dots \\ &= ap^{r-1} + ap^{r-2} + ap^{r-3} + \dots + a \\ &= a\tfrac{p^r-1}{p-1}.\end{aligned}$$

If $s = ap^r + s'$ $(a < p,\ s' < p^r)$, then

$$\begin{aligned}\mathbf{m}_{s,p} &= \mathbf{m}_{ap^r+s',p} \\ &= \left\lfloor \frac{ap^r + s'}{p} \right\rfloor + \left\lfloor \frac{ap^r + s'}{p^2} \right\rfloor + \left\lfloor \frac{ap^r + s'}{p^3} \right\rfloor + \dots \\ &= ap^{r-1} + \left\lfloor \frac{s'}{p} \right\rfloor + ap^{r-2} + \left\lfloor \frac{s'}{p^2} \right\rfloor + ap^{r-3} + \left\lfloor \frac{s'}{p^3} \right\rfloor + \dots + a + \left\lfloor \frac{s'}{p^r} \right\rfloor \\ &= ap^{r-1} + ap^{r-2} + ap^{r-3} + \dots + a + \left\lfloor \frac{s'}{p} \right\rfloor + \left\lfloor \frac{s'}{p^2} \right\rfloor + \left\lfloor \frac{s'}{p^3} \right\rfloor + \dots \\ &= a\tfrac{p^r-1}{p-1} + \mathbf{m}_{s',p}.\end{aligned}$$

If we now write s in its base-p expansion,

$$s = a_r p^r + a_1 p + a_{r-1} p^{r-1} + \dots + a_0,$$

we can see that

$$\mathbf{m}_{s,p} = a_r \tfrac{p^r-1}{p-1} + a_{r-1} \tfrac{p^{r-1}-1}{p-1} + \dots + a_1.$$

Now if we wish to find a value of s such that $\mathbf{m}_{s,p} \geqslant m$ and $\mathbf{m}_{s-1,p} < m$, *i.e.* $s = \mathbf{s}_{p^m}$, we can calculate the largest power of p less than or equal to s by finding the largest r such that

$$1 + p + p^2 + \dots + p^{r-1} \leqslant m$$

or

$$\frac{p^r - 1}{p - 1} \leqslant m$$

$\Rightarrow$

$$p^r \leqslant m(p-1) + 1$$

$\Rightarrow$

$$r \log p \leqslant \log (mp - m + 1)$$

$\Rightarrow$

$$r \leqslant \frac{\log (mp - m + 1)}{\log p}.$$

Hence

$$r = \left\lfloor \frac{\log (mp - m + 1)}{\log p} \right\rfloor.$$

Having calculated r, the largest power of p less than or equal to s, we can calculate the largest multiple, c, of p^r less than or equal to s as

$$c = \left\lfloor \frac{m(p-1)}{p^r - 1} \right\rfloor$$

since each multiple of p^r in s will contribute $\frac{p^r-1}{p-1}$ to $\mathbf{m}_{s,p}$.

Thus, we know that

$$\mathbf{m}_{cp^r,p} = c\left(\tfrac{p^r-1}{p-1}\right) \leqslant m < \mathbf{m}_{(c+1)p^r,p}$$

hence

$$cp^r \leqslant \mathbf{s}_{p^m} < (c+1)p^r$$

and so for some $b < p^r$, $s = cp^r + b$.

The value of b is the smallest such that

$$\mathbf{m}_{b,p} \geqslant d,$$

where $d = m - c\left(\frac{p^r-1}{p-1}\right)$. *i.e.*

$$b = \mathbf{s}_{p^d}. \tag{5.6}$$

Hence

$$\mathbf{s}_{p^m} = cp^r + \mathbf{s}_{p^d}$$

with r, c, d as in conditions (5.5).

□

The value of $b = \mathbf{s}_{p^d}$ in equation (5.6) can be calculated by applying equation (5.4) to the remaining powers of p required in $s!$, namely $m - c\left(\frac{p^r-1}{p-1}\right)$.
This may of course incur further applications of equation (5.4), however these repetitions will terminate since the value of $r \in \mathbb{Z}$ is reduced each time, and cannot become less than zero.

We can use Theorem 5.2 to define a recursive procedure to calculate the smallest s such that $p^m | s!$. This can be seen in Procedure 5.1.

Procedure 5.1 Procedure to calculate the smallest s such that $p^m | s!$.

procedure **min_factorial** $(p^m \in \mathbf{P}) \to (\mathbb{N})$

if $m = 0$ then

return 0

else

$r := \lfloor \log(mp - m + 1)/\log(p) \rfloor$

$a := \lfloor m(p-1)/(p^r - 1) \rfloor$

$m' := m - a(p^r - 1)/(p-1)$

return $ap^r +$ **min_factorial**$(p^{m'})$

end if

Examples

The smallest s such that 5^{27} divides $s!$ is 115.

Here, $p = 5$ and $m = 27$.

From Theorem 5.2 we put $m_1 = m$ and

$$r_1 = \left\lfloor \frac{\log(m_1 p - m_1 + 1)}{\log p} \right\rfloor = 2$$

$$c_1 = \left\lfloor \frac{m_1 p - m_1}{p^{r_1} - 1} \right\rfloor = 4$$

$$d_1 = m - c_1 \left(\frac{p^{r_1} - 1}{p - 1} \right) = 3$$

to yield

$$s = \mathrm{s}_{5^{27}} = 4 \times 5^2 + \mathrm{s}_{5^3}.$$

We can now use Theorem 5.2 to find $\mathbf{s}_{5^3}$ by putting $m_2 = d_1$ and

$$r_2 = \left\lfloor \frac{\log(m_2 p - m_2 + 1)}{\log p} \right\rfloor = 1$$
$$c_2 = \left\lfloor \frac{m_2 p - m_1}{p^{r_2} - 1} \right\rfloor = 3$$
$$d_2 = m - c_2 \left(\frac{p^{r_2} - 1}{p - 1} \right) = 0.$$

We now have

$$\mathbf{s}_{5^3} = 3 \times 5^1 + \mathbf{s}_{5^0}.$$

$\mathbf{s}_{5^0} = 0$ and so

$$s = \mathbf{s}_{5^{27}} = 4 \times 5^2 + 3 \times 5 = 115.$$

We can check that this value is correct by calculating

$$\mathbf{m}_{115,5} = \left\lfloor \frac{115}{5} \right\rfloor + \left\lfloor \frac{115}{5^2} \right\rfloor + \left\lfloor \frac{115}{5^3} \right\rfloor + \ldots = 27$$

and

$$\mathbf{m}_{114,5} = \left\lfloor \frac{114}{5} \right\rfloor + \left\lfloor \frac{114}{5^2} \right\rfloor + \left\lfloor \frac{114}{5^3} \right\rfloor + \ldots = 26.$$

Hence $\mathbf{s}_{5^{27}} = 115$ as evaluated by Procedure 5.1.

The smallest s such that 11^{86039} divides $s!$ is 860420.

Here, $p = 11$ and $m = 86039$.

From Theorem 5.2 we put $m_1 = m$ and

$$r_1 = \left\lfloor \frac{\log(86039p - 86038)}{\log p} \right\rfloor = 5$$
$$c_1 = \left\lfloor \frac{86039(p-1)}{p^5 - 1} \right\rfloor = 5$$
$$d_1 = 86039 - 5\left(\frac{p^5 - 1}{p - 1}\right) = 5514$$
$$\mathbf{s}_{p^{86039}} = 5p^5 + \mathbf{s}_{p^{5514}}.$$

To calculate $\mathbf{s}_{p^{5514}}$ we put $m_2 = 5514$ to yield

$$r_2 = \left\lfloor \frac{\log(5514p - 5513)}{\log p} \right\rfloor = 4$$
$$c_2 = \left\lfloor \frac{5514(p-1)}{p^4 - 1} \right\rfloor = 3$$
$$d_2 = 5514 - 3\left(\frac{p^4 - 1}{p - 1}\right) = 1122$$
$$\mathbf{s}_{p^{5514}} = 3p^4 + \mathbf{s}_{p^{1122}}.$$

Similarly, we now put $m_3 = 1122$ and

$$r_3 = \left\lfloor \frac{\log(1122p - 1121)}{\log p} \right\rfloor = 3$$
$$c_3 = \left\lfloor \frac{1122(p-1)}{p^3 - 1} \right\rfloor = 8$$
$$d_3 = 1122 - 8\left(\frac{p^3 - 1}{p - 1}\right) = 58$$
$$\mathbf{s}_{p^{1122}} = 8p^3 + \mathbf{s}_{p^{58}}.$$

Continuing the procedure, we put $m_4 = 58$ to give

$$r_4 = \left\lfloor \frac{\log(58p - 57)}{\log p} \right\rfloor = 2$$
$$c_4 = \left\lfloor \frac{58(p-1)}{p^2 - 1} \right\rfloor = 4$$
$$d_4 = 58 - 4\left(\frac{p^2 - 1}{p - 1}\right) = 10$$
$$\mathbf{s}_{p^{58}} = 4p^2 + \mathbf{s}_{p^{10}}.$$

Finally, we calculate with $m_5 = 10$ to give

$$r_5 = \left\lfloor \frac{\log(10p - 9)}{\log p} \right\rfloor = 1$$
$$c_5 = \left\lfloor \frac{10(p-1)}{p^1 - 1} \right\rfloor = 10$$
$$d_5 = 10 - 10\left(\frac{p^1 - 1}{p - 1}\right) = 0$$
$$\mathbf{s}_{p^{10}} = 10p + \mathbf{s}_{p^0} = 10p.$$

This tells us that

$$s = \mathbf{s}_{11^{86039}} = 5 \times 11^5 + 3 \times 11^4 + 8 \times 11^3 + 4 \times 11^2 + 10 \times 11 = 860429$$

We can check that this value is correct by calculating

$$\mathbf{m}_{860420,11} = \left\lfloor \frac{860420}{11} \right\rfloor + \left\lfloor \frac{860420}{11^2} \right\rfloor + \left\lfloor \frac{860420}{11^3} \right\rfloor + \ldots = 86039$$

and

$$\mathbf{m}_{860419,11} = \left\lfloor \frac{860419}{11} \right\rfloor + \left\lfloor \frac{860419}{11^2} \right\rfloor + \left\lfloor \frac{860419}{11^3} \right\rfloor + \ldots = 86038.$$

The smallest s such that $61^{1000000}$ divides $s!$ is 60000088.

Here, $p = 61$ and $m = 1000000$.

We put $m_1 = m$ and

$$r_1 = \left\lfloor \frac{\log(1000000p - 1000000 + 1)}{\log p} \right\rfloor = 4$$

$$c_1 = \left\lfloor \frac{1000000(p-1)}{p^4 - 1} \right\rfloor = 4$$

$$d_1 = 1000000 - 4\left(\frac{p^4-1}{p-1}\right) = 76944$$

$$\mathrm{s}_{p^{1000000}} = 4p^4 + \mathrm{s}_{p^{76944}}.$$

Applying the procedure to $\mathrm{s}_{p^{76944}}$ gives

$$r_2 = \left\lfloor \frac{\log(76944p - 76944 + 1)}{\log p} \right\rfloor = 3$$

$$c_2 = \left\lfloor \frac{76944(p-1)}{p^3 - 1} \right\rfloor = 20$$

$$d_2 = 76944 - 20\left(\frac{p^3-1}{p-1}\right) = 1284$$

$$\mathrm{s}_{p^{76944}} = 20p^3 + \mathrm{s}_{p^{1284}},$$

to $\mathrm{s}_{p^{1284}}$ gives

$$r_3 = \left\lfloor \frac{\log(1284p - 1284 + 1)}{\log p} \right\rfloor = 2$$

$$c_3 = \left\lfloor \frac{1284(p-1)}{p^2 - 1} \right\rfloor = 20$$

$$d_3 = 1284 - 20\left(\frac{p^2-1}{p-1}\right) = 44$$

$$\mathrm{s}_{p^{1284}} = 20p^2 + \mathrm{s}_{p^{44}},$$

and to $\mathbf{s}_{p^{44}}$ gives

$$\begin{aligned} r_4 &= \left\lfloor \frac{\log(44p-44+1)}{\log p} \right\rfloor = 1 \\ c_4 &= \left\lfloor \frac{44(p-1)}{p^1-1} \right\rfloor = 44 \\ d_4 &= 44 - 44\left(\frac{p^1-1}{p-1}\right) = 0 \\ \mathbf{s}_{p^{44}} &= 44p^1 + \mathbf{s}_{p^0}. \end{aligned}$$

Which means that

$$s = \mathbf{s}_{p^{1000000}} = 4 \times 61^4 + 20 \times 61^3 + 20 \times 61^2 + 44 \times 61 = 60000088$$

We can check that this value is correct by calculating

$$\mathbf{m}_{60000088,61} = 1000000$$

and

$$\mathbf{m}_{60000087,61} = 999999.$$

Hence $\mathbf{s}_{61^{1000000}} = 60000088$ as evaluated by Procedure 5.1.

The last two examples shown take the maximum number of iterations of applying Theorem 5.2 for values of p and m the size given, since $c_i \neq 0$ for $i = 1, 2, \ldots r_1$. However, we can see that the number of iterations of the procedure is substantially smaller than the size of p^m. Indeed, the maximum number of iterations can only be as large as r_1, since $0 < r_{i+1} < r_i$. Hence the number of iterations is of the

order of magnitude of

$$\begin{aligned} r_1 &= \left\lfloor \frac{\log m(p-1)}{\log p} \right\rfloor \\ &\leqslant \frac{\log m(p-1)}{\log p} \\ &= \frac{\log m + \log (p-1)}{\log p} \\ &< \frac{\log m}{\log p} + 1 \end{aligned}$$

Generating a Basis for $\mathcal{Z}_{p^m,x}$

In Procedure 5.1 we have an efficient method for calculating s_{p^r}. We can now proceed with the construction of a basis for $\mathcal{Z}_{p^m,x}$.

We are mainly interested in the zero evaluating polynomials

$$z_{p^r}(x) = x(x-1)\cdots(x-(\mathrm{s}_{p^r}-1))$$

which we now show are lowest degree monic polynomials such that $z(j) \equiv 0 \bmod p^r$ for $j \in \mathbb{Z}$.

LEMMA 5.3: The polynomial $x(x-1)\cdots(x-(\mathrm{s}_{p^r}-1))$ is a smallest degree monic polynomial in $\mathcal{Z}_{p^r,x}$. For a monic polynomial $z(x) \in \mathcal{Z}_{p^r,x}$ of this degree, s_{p^r}, there is a set of p values, $\{a_0, a_1, a_2, \ldots, a_{p-1}\} \subset \mathbb{Z}$, such that $p^r || z(a_i)$ $(i = 0, 1, 2, \ldots, p-1)$, and $a_i \equiv i \bmod p$.

PROOF: Firstly, we note that there may be more than one smallest degree monic polynomial in $\mathcal{Z}_{p^r,x}$. We only wish to show that that degree is s_{p^m}.

We label the statement in Lemma 5.3 as $S_{p,r}$.

We show that $S_{p,1}$ is true.
Because p is prime, the ring $\mathbb{Z}_p$ is a field, and hence it is well known that any polynomial $f(x) \in \mathbb{Z}_p[x]$ with distinct roots at $a_0, a_1, \ldots, a_k \in \mathbb{Z}_p$ must have $(x-a_0)(x-a_1)\cdots(x-a_k)$ as a factor.
Hence, any monic polynomial $f \in \mathbb{Z}[x]$ that evaluates to 0 modulo p for all $j \in \mathbb{Z}_p$ must have $x(x-a_0)(x-a_1)\cdots(x-a_{p-1})$, with $a_i \equiv i \bmod p$, as a factor and so must be of degree greater than or equal to p. Since $x(x-1)(x-2)\cdots(x-(p-1))$ is such a polynomial, it is of smallest degree.
Since $\mathrm{s}_{p^1} = p$, the first part of $S_{p,r}$ holds for $r = 1$.
The set of p values mentioned in the second part of $S_{p,r}$ can be $\{a_0+p, a_1+p, \ldots, a_{p-1}+p\}$.
We now assume that both parts of $S_{p,r}$ holds for $r \leqslant m-1$.

If $\mathrm{s}_{p^m} = \mathrm{s}_{p^{m-1}}$, then clearly $S_{p,m}$ holds, since the existence of a monic polynomial of degree less than s_{p^m} that always evaluates to zero modulo p^m contradicts $S_{p,m-1}$ which states that $x(x-1)\cdots(x-(\mathrm{s}_{p^{m-1}}-1))$, of degree s_{p^m} is a monic polynomial of smallest degree that always evaluates to zero modulo p^{m-1}.

If $\mathrm{s}_{p^m} \neq \mathrm{s}_{p^{m-1}}$, then $\mathrm{s}_{p^m} = \mathrm{s}_{p^{m-1}} + p$ (since $(\mathrm{s}_{p^{m-1}}+p)!$ has at least one more factor of p). Let us assume that $z(x) \in \mathbb{Z}[x]$ is a monic polynomial of smallest degree, s,

that evaluates to zero modulo p^m for all $x \in \mathbb{Z}$. By our previous assumption we know that $s \geqslant \mathbf{s}_{p^{m-1}}$. We also know that since $z(j) \equiv 0 \bmod p^m$, $z(j) \equiv 0 \bmod p$ for all $j \in \mathbb{Z}$, hence

$$\begin{aligned} z(x) &= q_1(x) f_1(x) \\ &= q_1(x)(x - a_{1,0})(x - a_{1,1}) \cdots (x - a_{1,p-1}) \end{aligned}$$

for some $a_{1,i} \equiv i \bmod p$, $i = 0, 1, 2, \ldots, p$.

Now the factor $f_1(x) = (x-a_{1,0})(x-a_{1,1}) \cdots (x-a_{1,p-1})$ of $z(x)$ only 'guarantees' a factor of p in $z(j)$ for $j = 0, 1, 2, \ldots, p^m - 1$, and it is not difficult to find a set of p values, $Z_{p,1}$, in $\mathbb{Z}$ that cover the set $\mathbb{Z}_p$ when evaluated modulo p, such that $p||f(j)$ for $j \in Z_{p,1}$. Such a set in this first instance is

$$Z_{p,1} = \{a_{1,0} + p, a_{1,1} + p, a_{1,2} + p, \ldots, a_{1,p-1} + p\}.$$

This means that $p^{m-1}|q_1(j)$ and in particular, $q_1(j) \equiv 0 \bmod p$, for $j \in Z_{p,1}$, hence

$$\begin{aligned} q_1(x) &= q_2(x) f_2(x) \\ &= q_2(x)(x - a_{2,0})(x - a_{2,1}) \cdots (x - a_{2,p-1}), \end{aligned}$$

i.e.

$$z(x) = q_2(x) f_1(x) f_2(x),$$

Now $f_1(x)f_2(x)$ is a polynomial of degree $2p$ and by our assumption above, can only guarantee a factor of $p^{\mathbf{m}_{2p,p}}$ ($\mathbf{m}_{2\cdot 2,2} = 3$, $\mathbf{m}_{2p,p} = 2$ for $p > 2$).
Hence there is another set of p numbers, $Z_{p,2} \subset \mathbb{Z}$, that covers $\mathbb{Z}_p$ when evaluated modulo p such that $p^{\mathbf{m}_{2p,p}}||f_1(j)f_2(j)$. For these values, $p^{m-\mathbf{m}_{2p,p}}|q_2(x)$ and

in particular, $q(j) \equiv 0 \bmod p$ for $j \in Z_{p,2}$. Hence

$$\begin{aligned} q_2(x) &= q_3(x)f_3(x) \\ &= q_3(x)(x - a_{3,0})(x - a_{3,1}) \cdots (x - a_{3,p-1}), \end{aligned}$$

and we continue in this way until we have

$$z(x) = q_s(x)f_1(x)f_2(x) \cdots f_s(x)$$

where $f_1(x)f_2(x) \cdots f_s(x)$ is of degree $\mathrm{s}_{p^{m-1}}$ and hence by our assumption can only guarantee a factor of p^{m-1} in $z(j)$ for $j \in \mathbb{Z}$. There is also a set $Z_{p,s}$ such that $p^{m-1} || f_1(j)f_2(j) \cdots f_s(j)$ for $j \in Z_{p,s}$. This means that there are values $a_{s,i} \in \mathbb{Z}$ with $a_{s,i} \equiv i \bmod p$ $(i = 0, 1, \ldots, p-1)$, such that

$$\begin{aligned} q_s(x) &= q(x)(x - a_{s,0})(x - a_{s,1}) \cdots (x - a_{s,p-1}) \\ &= q(x)f(x). \end{aligned}$$

We now have that

$$\begin{aligned} \deg z &= \deg q + \deg f_1 + \deg f_2 + \ldots + \deg f_s + \deg f \\ &= \deg q + \mathrm{s}_{p^{m-1}} + p \\ &= \deg q + \mathrm{s}_{p^m} \\ &\geqslant \mathrm{s}_{p^m}. \end{aligned}$$

Since $x(x-1) \cdots (x-(\mathrm{s}_{p^m} - 1))x$ is of degree s_{p^m}, $\deg q = 0$ and indeed $q(x) = 1$ because z is monic.

□

For $\mathcal{Z}_{p^m,x}$, we can use as a generating set the polynomials

$$\left\langle p^{m-r} z_{p^r}(x) \right\rangle$$

Each of these represents a polynomial of lowest degree with a content of p^r $(r = 0, 1, 2, \ldots, m)$ that evaluates to zero modulo p^m. *i.e.* , for each of the monic polynomials $z_{p^r}(x)$ $(r = 0, 1, 2, \ldots, m)$ we introduce an extra factor of a power of p to make up the power of p to m. We should note, that some of these polynomials are simply scalar multiples of a lesser poynomial (*e.g.* if we were working in $\mathcal{Z}_{24,x}$, $2^2 z_{2^2}(x)$ is twice $2z_{2^3}(x)$ since $z_{2^2}(x) = z_{2^3}(x)$ — because $s_{2^2} = s_{2^3}$). To take account of this, we express the set of generators in a different form to take account of these linearly dependent polynomials.

Instead of listing polynomials with degree s_{p^j}, which may repeat, we list them with degrees $p, 2p, 3p, \ldots, s_{p^m}$, being careful to multiply by a suitable power of p. Hence we can use

$$\langle p^{m-m_{cp,p}} x(x-1)\cdots(x-(cp-1)) | c = 0, 1, 2, \ldots, \tfrac{s_{p^m}}{p} \rangle.$$

as a basis for $\mathcal{Z}_{p^m,x}$.

5.4 Reduction of $f(x)$

This means that to see whether any given polynomial $f(x) \in \mathbb{Z}[x]$ is in a particular zero evaluating ideal, $\mathcal{Z}_{p^m,x}$, we must *reduce* the polynomial to its canonical representative within $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$. Since the representative of any $z \in \mathcal{Z}_{p^m,x}$ will be 0, $f \in \mathcal{Z}_{p^m,x}$ if and only if its representative is 0.

We reduce any polynomial $f(x) \in \mathbb{Z}[x]$ by using the fact that

$$f(x) = q_1(x) z_{p^m}(x) + r_1(x)$$

for some $q_1, r_1 \in \mathbb{Z}[x]$ with the degree of r_1 strictly less than s_{p^m}, the degree of $z_{p^m}(x)$. This means that f is composed of a certain amount (q_1) of the monic poly-

nomial $z_{p^m}(x)$ plus a remainder (r_1) that is of degree less than $z_{p^m}(x)$.

In terms of the PGPs created from f and r_1, they are identical. This is simply a consequence of the fact that the values of f modulo p^m are identical to the values of r_1 modulo p^m. We can see this since

$$\begin{aligned} r_1(j) &= f(j) - q_1(j) z_{p^m}(j) \\ &\equiv f(j) - 0 \bmod p^m \\ &\equiv f(j) \bmod p^m. \end{aligned}$$

Now we can reduce $f(x)$ further by 'factoring out' the next lowest degree basis polynomial of $\mathcal{Z}_{p^m,x}$, namely $pz_{p^{m-1}}(x)$, from $r_1(x)$. We should note that r_1 was guaranteed to be of degree less than s_{p^m} because $z_{p^m}(x)$ is a monic polynomial. However, any further reductions of f are not guaranteed to reduce its degree since the basis elements of degree less than s_{p^m} are not monic.
What we do achieve is to reduce the size of the coefficients of the exponents of degree greater than or equal to the degree of $pz_{p^{m-1}}(x)$, *i.e.* $\mathrm{s}_{p^{m-1}}$. These coefficients can be reduced to less than p, since that is the leading coefficient of $pz_{p^m-1}(x)$. We thus calculate

$$r_1(x) = q_2(x) \cdot pz_{p^{m-1}}(x) + r_2(x),$$

such that if $r_2(x) = a_0 + a_1 x + \ldots + a_s x^s$ (for some $s < \mathrm{s}_{p^m}$), then $a_i < p$ if $i \geqslant \mathrm{s}_{p^{m-1}}$.

This process can be continued until we are left with a final remainder

$$r_m(x) = b_0 + b_1 x + b_2 x^2 + \ldots + b_s x^s$$

such that

$$b_i < p^{m-k} \quad \text{if } i \geqslant \mathbf{s}_{p^k} \quad \text{for } k = 0, 1, 2, \ldots, m.$$

In practice when calculating r_k by reducing r_{k+1} it is easier to reduce r_{k+1} by one exponent at a time, *e.g.* when reducing $\bar{f}(x) = f(x) = a_0 + a_1 x + \ldots + a_s x^s$ to r_1 we calculate

$$\begin{aligned} r_{1,1}(x) &= \bar{f}(x) - a_s \times z_{p^m}(x) x^{s - \mathbf{s}_{p^m}} \\ &= b_0 + b_1 x + \ldots + b_{s-1} x^{s-1} \end{aligned}$$

which is a polynomial of degree (at least) one less than f. If we then redefine $\bar{f}$ as $r_{1,1}$ (which evaluates to the same sequence modulo p^m) and calculate $r_{1,2}$ in the same way from this new $\bar{f}$, we again reduce the degree, but do not effect the PGP genereted from this polynomial. We can continue this process until $\bar{f}$ is of degree $\mathbf{s}_{p^m} - 1$, one less than the degree of $z_{p^m}(x)$.

We could then continue to reduce the coefficients of $\bar{f}$ by subtracting as many multiples of $p z_{p^{m-1}}(x)$ as we need to reduce the coefficients of powers between $\mathbf{s}_{p^{m-1}}$ and $\mathbf{s}_{p^m}$ to a value less than p.

Carrying on such a process reduces $\bar{f}$ such that the coefficients of powers between $\mathbf{s}_{p^k}$ and $\mathbf{s}_{p^{k+1}}$ are between 0 and $p^{m-k} - 1$ inclusive.
Procedure 5.2 defines exactly such a method.

Procedure 5.2 Reduces $f(x) \in \mathbb{Z}[x]$ to its canonical representative $\bar{f}(x) \in \mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$. $\bar{f}$ is such that $\bar{f}(j) \equiv f(j) \bmod p^m \quad (j = 0, 1, 2, \ldots)$

procedure **reduce** $(f \in \mathbb{Z}[x], p \in \mathbf{P}) \rightarrow (\mathbb{Z}[x]/\mathcal{Z}_{p^m,x})$

$\bar{f}(x) := f(x)$

$\sum_{j=0}^{r} a_j x^j := \bar{f}(x)$

$i := r$

for $j := 0$ to m

$s :=$ **min_factorial**(p^{m-j})

for $k := 0$ to $i - s$

$\sum_{j=0}^{r'} a_j x^j := \bar{f}(x)$

$\bar{f}(x) := \bar{f}(x) - p^j \times \lfloor a_{i-k}/p^j \rfloor \times x^{i-k-s} x(x-1)\cdots(x-(s-1))$

end for

$i := s$

end for

return $\bar{f}(x)$

Examples

The canonical representative of $f(x) = 10x^{10} + 9x^9 + 8x^8 + 7x^7 + 6x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x$ in $\mathcal{Z}_{2^3,x}$ is $\bar{f}(x) = 2x^2 + 5x$

Applying Procedure 5.2 we start by setting

$\bar{f}(x) = 10x^{10} + 9x^9 + 8x^8 + 7x^7 + 6x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x$ and $i = 10$.

We then start the first pass of the outer loop with j taking a value from 0 to 2 with $j = 0$.

We put

$s = s_{2^3} = 4$

and start the first pass of the inner loop with k taking on values from 0 to 6 starting with

$k = 0$.

$\bar{f}(x)$ becomes $\bar{f} - \lfloor \frac{10}{1} \rfloor x^6 x(x-1)(x-2)(x-3)$

i.e. $\bar{f}(x) = 69x^9 - 102x^8 + 67x^7 + 6x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x.$

The second pass gives

$k = 1$ and

$\bar{f}(x) = 312x^8 - 692x^7 + 420x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x.$

Subsequent passes of the inner loop give

$k = 2$ and

$\bar{f}(x) = 1180x^7 - 3012x^6 + 1877x^5 + 4x^4 + 3x^3 + 2x^2 + x.$

$k = 3$ and

$\bar{f}(x) = 4068x^6 - 11103x^5 + 7084x^4 + 3x^3 + 2x^2 + x.$

$k = 4$ and

$\bar{f}(x) = 13305x^5 - 37664x^4 + 24411x^3 + 2x^2 + x.$

$k = 5$ and

$\bar{f}(x) = 42166x^4 - 121944x^3 + 79832x^2 + x.$

$k = 6$ and

$\bar{f}(x) = 131052x^3 - 383994x^2 + 252997x$

which concludes the inner loop. We then set

$i = 4$

to conclude the first pass of the outer loop.

The second pass of the outer loop sets

$j = 1$ and $s = 4$.

Note that the new value of s is the same as the old value of s, hence the new inner loop takes k from 0 to 0 and does not have any change on $\bar{f}(x)$.

The third pass of the outer loop sets

$j = 2$.

We calculate

$s = 2$.

The inner loop is then restarted with k going from 0 to $i - s = 2$ starting with

$k = 0$. We then set

$\bar{f}(x) = 131052x^3 - 383994x^2 + 252997x.$

Subsequent passes of the inner loop give $k = 1$ and

$\bar{f}(x) = -252942x^2 + 252997x.$

$k = 2$ and

$\bar{f}(x) = 2x^2 + 53x.$

which concludes the inner loop, and we set

$i = 2.$

The fourth pass of the outer loop sets

$j = 3$. Since $j = m$ this will be the last pass of the outer loop and the procedure will terminate.

We set

$s = 0.$

We start the inner loop once more with

$k = 0$ and set

$\bar{f}(x) = 2x^2 + 53x.$

$k = 1$ and

$\bar{f}(x) = 2x^2 + 5x.$

$k = 2$ and

$\bar{f}(x) = 2x^2 + 5x$

conludes the inner loop.

The procedure now returns its latest value of $\bar{f}(x)$, which is $2x^2 + 5x$ as the canonical representative for $10x^{10} + 9x^9 + 8x^8 + 7x^7 + 6x^6 + 5x^5 + 4x^4 + 3x^3 + 2x^2 + x$ modulo 2^3.

The canonical representative of $f(x) = x^{10} + x^9 + 8x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$ in $\mathcal{Z}_{3^2,x}$ is $\bar{f}(x) = x^4 + x^3 + 4x^2 + 4x + 1$

Applying Procedure 5.2 we start by setting

$\bar{f} = x^{10} + x^9 + 8x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$ and $i = 10$.

We then start the outer loop with

$j = 0$

We set

$s = 6$.

The inner loop is then started for the first time with $k = 0$ and we set

$\bar{f}(x) = 16x^9 - 84x^8 + 226x^7 - 273x^6 + 121x^5 + x^4 + x^3 + x^2 + x + 1$.

Subsequent passes of the inner loop give $k = 1$ and

$\bar{f}(x) = 156x^8 - 1134x^7 + 3327x^6 - 4263x^5 + 1921x^4 + x^3 + x^2 + x + 1$.

$k = 2$ and

$\bar{f}(x) = 1206x^7 - 9933x^6 + 30837x^5 - 40823x^4 + 18721x^3 + x^2 + x + 1.$

$k = 3$ and

$\bar{f}(x) = 8157x^6 - 71673x^5 + 230527x^4 - 311723x^3 + 144721x^2 + x + 1.$

$k = 4$ and

$\bar{f}(x) = 50682x^5 - 462818x^4 + 1523602x^3 - 2090297x^2 + 978841x + 1.$

We set

$i = 6$ to conclude the first run of the inner loop.

The second pass of the outer loop now has

$j = 1.$

Calculate

$s = 3$

Start the inner loop with

$k = 0$ and calculate

$\bar{f}(x) = 50682x^5 - 462818x^4 + 1523602x^3 - 2090297x^2 + 978841x + 1.$

$k = 1$ and

$\bar{f}(x) = -310772x^4 + 1422238x^3 - 2090297x^2 + 978841x + 1.$

$k = 2$ and

$\bar{f}(x) = x^4 + 489919x^3 - 1468751x^2 + 978841x + 1.$

$k = 3$ and

$\bar{f}(x) = x^4 + x^3 + 1003x^2 - 995x + 1.$

We now set

$i = 3$ to end the inner loop.

The third pass of the outer loop has

$j = 2.$

We set

$s = 0$

and start the inner loop for the last time with

$k = 0$ and calculate

$\bar{f}(x) = x^4 + x^3 + 1003x^2 - 995x + 1.$

Continuing with

$k = 1$ and

$\bar{f}(x) = x^4 + x^3 + 4x^2 - 995x + 1.$

$k = 2$ and

$\bar{f}(x) = x^4 + x^3 + 4x^2 + 4x + 1.$

$k = 3$ and

$\bar{f}(x) = x^4 + x^3 + 4x^2 + 4x + 1.$

We then set

$i = 0$ signalling the end of the procedure

which returns $\bar{f}(x) = x^4 + x^3 + 4x^2 + 4x + 1$ as the canonical representative of $f(x) = x^{10} + x^9 + 8x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$ in $\mathcal{Z}_{3^2,x}$.

5.5 Determining membership of $\mathcal{Z}_{p^m,x}$

To determine whether a given polynomial, $f(x)$ is in $\mathcal{Z}_{p^m,x}$, we simply need to calculate its reduced polynomial, $\bar{f}(x)$, by applying Procedure 5.2. $f \in \mathcal{Z}_{p^m,x}$ if and only if $\bar{f} = 0$. We can now implement the procedure **in_Z** first mentioned in Section 3.2 as Procedure 5.3.

Procedure 5.3 Calculates whether $f(x) \in \mathcal{Z}_{p^m,x}$

```
procedure in_Z (f ∈ Z[x], p^m ∈ P) → (BOOLEAN)
    f̄(x) := reduce(f(x), p^m)
    if f̄(x) = 0 then
        return TRUE
    else
        return FALSE
    end if
```

Examples

$x^6 + 7x^4 + x^2 \in \mathcal{Z}_{3^2,x}$.

Putting $f(x) = x^6 + 7x^4 + x^2$ and $p^m = 3^2$ we run the procedure **reduce**$(f(x), p^m)$ to see if the canonical representative, $\bar{f}$, of f in $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$ is 0.

With $j = 0$ in the first pass of the loop, we get $s = 6$

$k = 0$ and

$\bar{f}(x) = 15x^5 - 78x^4 + 225x^3 - 273x^2 + 120x$

We put $i = 6$ and enter the second pass of the loop

$j = 1$ and $s = 3$

k now loops thorugh a number of values and the degree of $\bar{f}$ is reduced each time:

$k = 0$

$\bar{f}(x) = 15x^5 - 78x^4 + 225x^3 - 273x^2 + 120x$

$k = 1$

$\bar{f}(x) = -33x^4 + 195x^3 - 273x^2 + 120x$

$k = 2$

$\bar{f}(x) = 96x^3 - 207x^2 + 120x$

$k = 3$

$\bar{f}(x) = 81x^2 - 72x.$

We put $i = 3$.

In the third pass we have $j = 2$ and so $s = 0$

Again the inner loop is passed with a range of values of k, this time reducing the coefficients of $\bar{f}$.

$k = 0$

$\bar{f}(x) = 81x^2 - 72x$

$k = 1$

$\bar{f}(x) = -72x$

$k = 2$

$\bar{f}(x) = 0$

$k = 3$

$\bar{f}(x) = 0$

We now have $i = 0$ and so our final value of $\bar{f}$ is 0, hence $x^6 + 7x^4 + x^2 \in \mathcal{Z}_{3^2,x}$.

Number of Sequences Generated by Integer Coefficient Polynomials modulo p^m

Calculating the size of $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$, *i.e.* the number of different canonical representatives that can be output by **reduce** for any given p^m, gives us the number of *different* sequences that can be generated by integer-coefficient polynomials evaluated modulo p^m.

This number can also be thought of as the size of $\mathrm{Im}\,\psi$ where ψ is the homomorphism defined in equation (5.1) at the end of Section 5.2.

Chapter 6

Unbounded Symmetries

As discussed in Section 1.4, the symmetry groups of the unbounded PGPs are those of the seven frieze groups shown in Figure 1.3 on page 11. The symmetries of these groups are a translation (T), which is always present in an unbounded PGP, a rotation (R) of $\frac{\pi}{2}$, a reflection (V) in a line perpendicular to the translation, a reflection (H) in a line parallel to the translation and a glide reflection (G) – a combination of a translation and reflection in a line parallel to the translation. We refer to unbounded PGPs as *open*.

6.1 Detecting Frieze Symmetry Generators

The ideas behind detecting the symmetries of an open PGP are much the same as those behind detecting the symmetries of closed PGPs. We will look at various differences of the generating polynomial f and see whether the result is a zero evaluating ideal. If so, then the angular difference of various pairs of edges can be determined that in turn will imply a symmetry.

To detect whether the PGP generated from a given $f \in \mathbb{Z}[x]$ and $p^m \in \mathbf{P}$ is unbounded we can use procedure **is_closed** (Procedure 4.1 on page 38).

Translation

The translational symmetry of an unbounded PGP is certain. The vertex at the first repetition of the pattern marks the point whose position vector corresponds to the translation vector. The point of first repetition can be found using the same procedure as for bounded PGPs, **first_rep** (Procedure 3.1 on page 30). We can use this point to limit the number of points checked in a search for reflectional and rotational symmetries.

Reflection in Line Perpendicular to Translation

What we are checking for here is the same kind of reflectional symmetry detected in closed PGPs. That is, the edges equally distant on either side of a particular vertex (or mid-point of an edge) are oriented with one at the negative angle of the other (plus $\frac{\pi}{2}$ since one is mirrored in the 'opposite direction' to the other) relative to the line bisecting the edges adjacent to the vertex or edge in question (see Figures 4.1 on page 59 and 4.2 on page 62). We can thus use the procedure **has_ref_symm** (Procedure 4.3 on page 65), although we should note that the calculation of $\rho :=$ **rot_symm**(f, p^m) is unecessary since the bounded rotational symmetry is guaranteed not to exist! The procedure **rot_symm**(f, p^m) will in this case return the value 1. This prompts us to create an alternative procedure identical to Procedure 4.3 with the exception of the calculation of ρ mentioned above. We shall call this procedure **open_has_ref_symm** shown in Procedure 6.1.

Procedure 6.1 Determines whether the open PGP $\mathcal{P}_{f(x),p^m}$ has a line of reflection.

```
procedure open_has_ref_symm (f ∈ ℤ[x], p^m ∈ P) → (BOOLEAN)
    p^m' := first_rep(f, p^m)
    r := p^m'
    for k := 0 to r − 1
        if in_Z(Δf(k + x) − Δf(k − 2 − x), p^m)
            OR in_Z(Δf(k − 1 + x) − Δf(k − x), p^m) then
            return TRUE
        end if
    end for
    return FALSE
```

Example

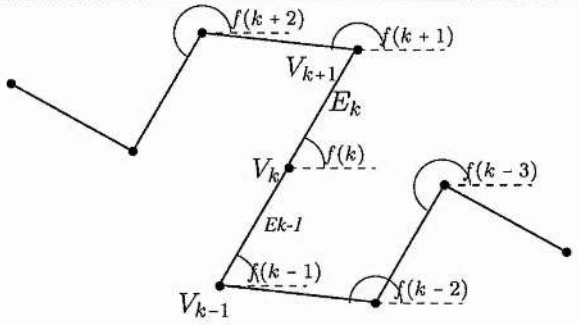
The open PGP $\mathcal{P}_{4x^3+13x^2+22^x+11,25}$ has reflectional symmetry.

With $f(x) = 4x^3 + 13x^2 + 22^x + 11$ and $p^m = 5^2$, the procedure first calculates $p^{m'}$ as 25 using the procedure **first_rep**(f, p^m). This means we have to check potentially 25 vertices and edge mid-points as the intersection with an axis of reflectional symmetry. Fortunately in this example we don't have to check all 25.

For $k = 0$, **in_Z**$(\Delta f(k + x) - \Delta f(k - 2 - x), p^m)$ returns FALSE and **in_Z**$(\Delta f(k - 1 + x) - \Delta f(k - x), p^m)$ returns FALSE .

For $k = 1$, **in_Z**$(\Delta f(k + x) - \Delta f(k - 2 - x), p^m)$ again returns FALSE , however

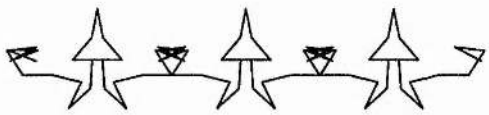
Figure 6.1 Rotational symmetry in an open PGP about a vertex.



in_Z$(\Delta f(k-1+x) - \Delta f(k-x), p^m)$ this time returns **TRUE** since

$$f(x) - f(-x) = 100x - 50 \equiv 0 \bmod 25.$$

Hence $\mathcal{P}_{4x^3+13x^2+22x+11,25}$ has reflectional symmetry.

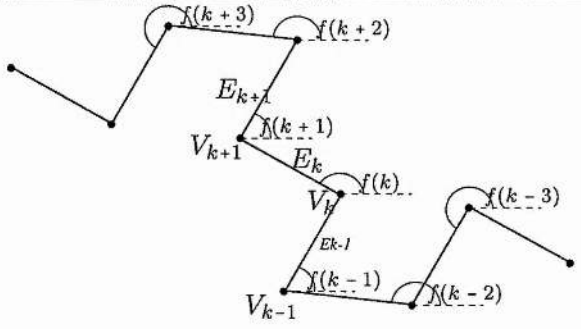


$\mathcal{P}_{4x^3+13x^2+22x+11,25}$

Rotational Symmetry

It is clear that since an open PGP contains a translational symmetry, the only rotational symmetry that might occur is of degree 2. The rotational symmetry found in an open PGP is different in nature to the rotational symmetry found in bounded PGPs. What we are looking for in this case are the edges equally distant on either side of a particular vertex (or mid-point of an edge) oriented the same (*i.e.* in opposite directions when considering one edge is being traversed in the 'opposite direction'). Figures 6.1 and 6.2 demonstrate these cases.

Figure 6.2 Rotational symmetry in an open PGP about the mid-point of an edge.



From Figure 6.1 we can see that rotational symmetry can occur around the vertex V_k if and only if

$$f(k+j) \equiv f(k-j-1) \bmod p^m$$

for $j = 0, 1, 2, \ldots$.

i.e.

$$f(k+x) - f(k-x-1) \in \mathcal{Z}_{p^m,x}. \tag{6.1}$$

From Figure 6.2 we can see that rotational symmetry can occur around the centre of the edge E_k if and only if

$$f(k+j) \equiv f(k-j) \bmod p^m$$

for $j = 0, 1, 2, \ldots$.

i.e.

$$f(k+x) - f(k-x) \in \mathcal{Z}_{p^m,x}. \tag{6.2}$$

We can use equations (6.1) and (6.2) in a procedure, **open_has_rot_symm**, that takes $f \in \mathbb{Z}[x]$ and $p^m \in \mathbf{P}$ and returns a boolean TRUE if the open PGP, $\mathcal{P}_{f,p^m}$, has rotational symmetry. This can be seen in Procedure 6.2.

Procedure 6.2 Determines whether the open PGP generated from $f \in \mathbb{Z}[x]$ and p^m has rotational symmetry of degree 2.

```
procedure open_has_rot_symm (f ∈ Z[x], p^m ∈ P) → (BOOLEAN)
    p^m' := first_rep(f, p^m)
    for k := 0 to p^m' − 1
        if in_Z(f(k+x) − f(k−x−1), p^m) OR in_Z(f(k+x) − f(k−x), p^m) then
            return TRUE
        end if
    end for
    return FALSE
```

Example

The open PGP $\mathcal{P}_{4x^6+2x^2,27}$ has rotational symmetry (of order 2).

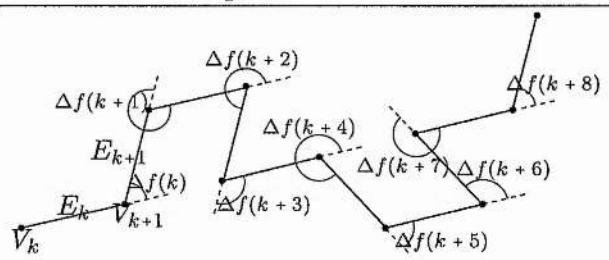
With $f(x) = 4x^6 + 2x^2$ and $p^m = 3^3$, the procedure first calculates $p^{m'}$ as 27 using the procedure **first_rep**(f, p^m). This means we have to check potentially 27 vertices and edge mid-points as the centre of rotational symmetry. Fortunately in this example we don't have to check all 27.

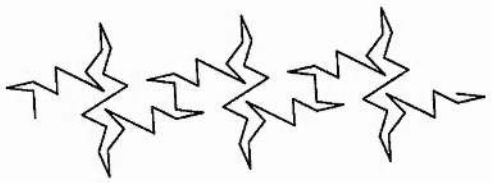
For $k = 0$, **in_Z**$(f(x) - f(-x - 1), p^m)$ returns FALSE , however,
in_Z$(f(x) - f(-x), p^m)$ returns TRUE since

$$f(x) - f(-x) = 0.$$

Hence $\mathcal{P}_{4x^6+2x^2,27}$ has rotational symmetry of order 2.

Figure 6.3 Glide reflection in an open PGP.





$\mathcal{P}_{4x^6+2x^2,27}$ (rotated for space)

Glide Reflection

The detection of a glide reflection is very close to the detection of the first repetition of a PGP given f and p^m. Since a glide reflection is a repetition of the initial pattern reflected in a line parallel to the translation, we would have a number of edges, followed by the same number of edges and then the first repeat would occur. This means that the point of first repetition must correspond to an even number, hence $p = 2$ for a glide reflection. Similarly, the reflected part must start at the point which corresponds to exactly half way to the point of first repetition.
From Figure 6.3 we can see that for a glide reflection to occur, the forward difference of $f(x)$ must satisfy

$$\Delta f(j) + \Delta f(j + \tfrac{r}{2}) \equiv 0 \bmod p^m$$

for $j = 0, 1, 2 \ldots$ and $r = \textbf{first\_rep}(f, p^m)$. From this we have

$$\Delta f(x) + \Delta f(x + \tfrac{r}{2}) \in \mathcal{Z}_{p^m,x}. \tag{6.3}$$

Equation (6.3) allows us to formulate the procedure **has_glide_ref** which takes $f \in \mathbb{Z}[x]$ and $p^m \in \mathbf{P}$ and returns a BOOLEAN value depending on whether the open $\mathcal{P}_{f,p^m}$ has a glide reflection.

Procedure 6.3 Determines whether the open PGP generated from $f \in \mathbb{Z}[x]$ and p^m has a glide reflection.

```
procedure has_glide_ref (f ∈ Z[x], p^m ∈ P) → (BOOLEAN)
    if p = 2 then
        p^m' := first_rep(f, p^m)
        if m' > 0 then
            if in_Z(Δf(x + p^(m'-1)) + Δf(x), p^m) then
                return TRUE
            end if
        end if
    end if
    return FALSE
```

Example

The open PGP $\mathcal{P}_{x^5+x^4+3x^3+5x^2+2,16}$ has a glide reflection.

With $f(x) = x^5 + x^4 + 3x^3 + 5x^2 + 2$ and $p^m = 2^4$, the procedure first calculates $p^{m'}$ as 4 using the procedure **first_rep**(f, p^m). Procedure 6.3 enters no additional loops, and so the number of tests does not depend on f or p^m.

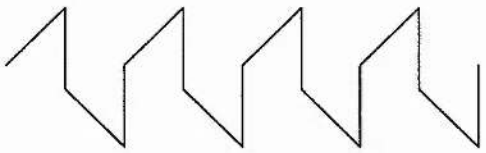
In this case, $m' > 0$ so we need to calculate whether

$$\Delta f(x + p^{m'-1}) + \Delta f(x) \in \mathcal{Z}_{p^m,x}$$

and

$$\Delta f(x + p^{m'-1}) + \Delta f(x) = 10x^4 + 188x^3 + 2306x^2 + 13384x + 29492.$$

The procedure **in_Z** now tells us that this polynomial is indeed in $\mathcal{Z}_{p^m,x}$ and hence $\mathcal{P}_{x^5+x^4+3x^3+5x^2+2,16}$ has a glide reflection.



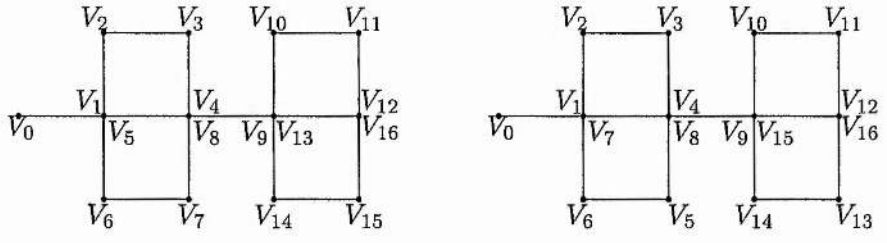
$\mathcal{P}_{x^5+x^4+3x^3+5x^2+2,16}$

6.2 Reflection in Line Parallel to Translation

When a PGP that displays a reflective symmetry in a line parallel to the translation is considered, we realise that such a reflection can only be achieved by means of a contrived manner. The order of the edges does not really lend itself to a 'sensible' map that takes an edge onto its image after reflection. This is because any such map will invariably miss the mapping of edges along the central line (the line of reflection) onto themselves, which would be desirable.

One way we can make an open PGP to have a reflection in a line parallel to the translation is to 'double back' for the reflective repetition (see Figure 6.5(a)). Alternatively, we might repeat the pattern from one side in reverse on the other, before

Figure 6.4 Possible methods for an open PGP construction to display reflective symmetry in a line parallel to the translational symmetry.

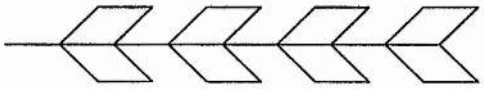


(a) Doubling back (b) Reverse reflection

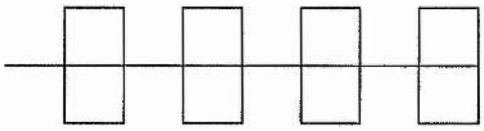
translating along the central line (see Figure 6.5(b)).

Since a reflecton in a line parallel to the translation can only be achieved in an 'unnatural' manner we do not present any methods to detect this symmetry. However, two examples are presented of PGPs that display a reflection in a line parallel to the translation. Both these examples were carefully constructed using the methods described later in Chapter 8.

Examples



$\mathcal{P}_{x^9+2x^7+25x^5+100x^3+64x^2+96x,256}$



$\mathcal{P}_{x^7+11x^3+8x^2+20x,32}$

Part III

Construction

Chapter 7

Unit Periodic Sequences

7.1 Introduction

In Part II we have determined how to predict the symmetry group of the shape of a PGP derived from a given polynomial evaluated modulo a given prime power. In this part we endeavour to determine how we might go about constructing a polynomial evaluated modulo a prime power which will produce a PGP with a desired (feasible) polygon.

Feasible Polygons

We define here what we mean by a *feasible polygon*. The polygon may be open or closed.

Clearly, any polygon that is to be derived from a PGP must have its edges of unit length, and each edge, E_j, restricted to an orientation of $a_j \frac{2\pi \mathrm{i}}{p^m}$ from the x-axis, for some $p^m \in \mathbf{P}$ and $a_j \in \mathbb{Z}_{p^m}$.
We must also stipulate that the pattern of edges must repeat after the $p^{m'}$-th edge, or close after the $p^{m'}$-th edge for some $0 \leqslant m' \leqslant m$.

DEFINITION 7.1: A *feasible polygon* for generation by a PGP is any polygon with edges of unit length, all restricted to an orientation that is a multiple of $\frac{2\pi}{p^m}$ radians from the horizontal, for some $p^m \in \mathbf{P}$, and *either* a) $p^{m'}$ edges, or b) a repetition after $p^{m'}$ edges, where $0 \leqslant m' \leqslant m$.

Any such feasible polygon has an associated p^m-periodic sequence of values from $\mathbb{Z}_{p^m}$, and any p^m-periodic sequence of integers has an associated feasible polygon.

Unit Periodic Sequences

We now introduce the concept of unit periodic sequences which can be used to construct any periodic sequence we might require to generate a given feasible polygon.

DEFINITION 7.2: The n *unit periodic sequences*, $U_{n,i}$ $(i = 0, 1, 2, \ldots, n-1)$, with period n, are defined as

$$U_{n,i} = (a_j)_j$$

with

$$a_j = \begin{cases} 1 & \text{if } j \equiv i \bmod n, \\ 0 & \text{otherwise.} \end{cases}$$

for $i = 0, 1, 2, \ldots, n-1$.

i.e.

$$U_{n,i} = (\overbrace{0,\ldots,0}^{i},1,\overbrace{0,\ldots,0}^{n-1},1,\overbrace{0,\ldots,0}^{n-1},\ldots)$$

for $i = 0, 1, 2, \ldots, n-1$.

We can use these unit periodic sequences to construct in a linear way any integer sequence, A, with period p^m:

$$A = (a_j)_j = \sum_{j=0}^{p^m-1} a_j U_{p^m,j}.$$

Iterated Forward Differences

To simplify several of the proofs later in this section, we start with a lemma that allows us to restrict our examination of the iterated forward differences on an arbitrary p^m-periodic integer sequence to that of just the sequence $U_{p^m,0}$.

LEMMA 7.1: If $\Delta^k U_{n,0} = (u_j)_j$, then

$$\Delta^k U_{n,i} = (u_{j-i+n})_j,$$

for $i = 0, 1, 2, \ldots, n-1$.

PROOF:

$$U_{n,0} = (1, \overbrace{0, \ldots, 0}^{n-1}, 1, 0, 0, \ldots)$$

$$\Delta^1 U_{n,0} = \Delta U_{n,0} = (-1, \overbrace{0 \ldots, 0}^{n-2}, 1, -1, 0, 0, \ldots)$$

$$U_{n,i} = (\overbrace{0, \ldots, 0}^{i}, 1, \overbrace{0, \ldots, 0}^{n-1}, 1, 0, 0, \ldots)$$

$$\Delta^1 U_{n,i} = \Delta U_{n,i} = (\overbrace{0, \ldots, 0}^{i}, 1, -1, \overbrace{0 \ldots, 0}^{n-2}, 1, -1, 0, 0, \ldots)$$

and so the lemma is true for $k = 1$.

It is clear that since U_{n,i_1} is the sequence $U_{n,0}$ with each term shifted $n - i_1$ places to the left (the terms with negative indices being lost), that U_{n,i_1+i_2} is the same sequence as U_{n,i_1} with each term shifted $n - i_2$ places to the left.

Assuming that the lemma is true for $k \leqslant r$, then since $\Delta^r U_{n,0}$ is a linear combination of the $U_{n,i}$ $(i = 0, 1, \ldots, n-1)$, say

$$\Delta^r U_{n,0} = \sum_{j=0}^{n-1} c_j U_{n,j},$$

then

$$\Delta^r U_{n,i} = \sum_{j=0}^{n-1} c_j U_{n,j+i}.$$

We now have that

$$\Delta^{r+1} U_{n,0} = \sum_{j=0}^{n-1} c_j \Delta U_{n,j}$$

and

$$\Delta^{r+1}U_{n,i} = \sum_{j=0}^{n-1} c_j \Delta U_{n,j+i}.$$

Hence, $\Delta^{r+1}U_{n,i}$ is the same linear combination of the shifted unit periodic sequences and is itself simply a version of $\Delta^{r+1}U_{n,0}$ with each term shifted $n-i$ places to the left.

□

We have established that any p^m-periodic sequence, $A = (a_j)_j$, can be written as the linear combination of unit sequences $U_{p^m,i}$ $(i = 0, 1, \ldots, p^m - 1)$. We have also shown that the results of iterating the forward difference operator, Δ, on each $U_{p^m,i}$ are essentially the same. This means that examining the behaviour of an arbitrary p^m-periodic integer sequence when iterating the forward difference operator, we need only examine the behaviour of the first unit p^m-periodic sequence, $U_{p^m,0}$.

7.2 Limits of Iterated Forward Differences

In this section we will establish that the limit, modulo p^m, of a p^m-periodic integer sequence when iterations of the forward difference operator are taken, is the zero sequence. The limit is attained (necessarily, since the sequences are of integer values) and we will prove in Theorem 7.4 that it is always attained by the $(p^m + (m-1)(p-1)p^{m-1})$-th iteration.

When we iterate the forward difference operator on $U_{p^m,0}$, the non-zero terms involve alternating sums of binomial coefficients, and so we will find it useful to prove the following:

LEMMA 7.2: If $k = p^{m-1}(p-1)$, then the binomial coefficient

$$\binom{k}{\ell} \equiv \begin{cases} (-1)^{qp} \bmod p & \text{if } \ell = qp^{m-1} \quad (0 \leqslant q < p), \\ 0 \bmod p & \text{otherwise.} \end{cases}$$

PROOF: We will prove this lemma in two ways. The first is a direct approach counting factors of p, whilst the second is an algebraic argument.

The direct approach starts by considering the binomial coefficient $b = \binom{k}{\ell}$ which can be written as a ratio of factorials:

$$\begin{aligned} b &= \binom{k}{\ell} \\ &= \frac{k!}{\ell!(k-\ell)!} \\ &= \frac{k \cdot (k-1) \cdot (k-2) \cdots (k-\ell+1)}{1 \cdot 2 \cdot 3 \cdots \ell}. \end{aligned} \tag{7.1}$$

Putting $k = (p-1)p^{m-1}$ and $\ell = qp^{m-1} + r \quad (0 \leqslant q < p, 0 \leqslant r < p^{m-1})$, we wish to evaluate (7.1) modulo p, since b is an integer and has a value modulo p. To find this value, we first evaluate each factor in the numerator and denominator of (7.1) modulo p^{m-1} if it is *not* a multiple of p^m, and as a multiple of p^{m-1} otherwise.

When $0 < r < p^{m-1}$ we get

$$\begin{aligned}&((p-1)p^{m-1}) \cdot (-1) \cdots (-p^m+1)\\ &\times((p-2)p^{m-1}) \cdot (-1) \cdots (-p^m+1)\\ &\times \cdots\\ &\times((p-1-q)p^{m-1}) \cdot (-1) \cdots (-r+1))\end{aligned}$$

in the numerator and

$$1 \cdot 2 \cdots (p^{m-1}) \cdot 1 \cdot 2 \cdots (2p^{m-1}) \cdots 1 \cdot 2 \cdots (qp^{m-1}) \cdot 1 \cdot 2 \cdots r$$

in the denominator.

From this it is clear that most of the terms that are multiples of p will cancel each other, and most of the none-multiples of p will also cancel leaving

$$b \equiv \frac{(-1)^{q(p^{m-1}-1)+r-1} \cdot ((p-1)p^{m-1}) \cdot ((p-2)p^{m-1}) \cdots (p-1-q)p^{m-1}}{p^{m-1} \cdot 2p^{m-1} \cdots qp^{m-1} \cdot r}. \qquad (7.2)$$

It is clear in this case where ℓ is not a multiple of p^m from equation (7.2) that there is an extra factor of p^{m-1} in the numerator than in the denominator, and so

$$b \equiv 0 \bmod p.$$

In the case where ℓ is a multiple of p^{m-1}, *i.e.* $\ell = qp^{m-1} \quad (0 < q < p)$, then the expression in (7.1) we will have

$$\begin{aligned}&((p-1)p^{m-1}) \cdot (-1) \cdots (-p^m+1)\\ &\times((p-2)p^{m-1}) \cdot (-1) \cdots (-p^m+1)\\ &\times \cdots\\ &\times((p-q)p^{m-1}) \cdot (-1) \cdots (-p^m+1)\end{aligned}$$

in the numerator and

$$1 \cdot 2 \cdots (p^{m-1})1 \cdot 2 \cdots (2p^{m-1}) \cdots 1 \cdot 2 \cdots (qp^{m-1})$$

in the denominator.

Now we have a different set of terms that are multiples of p cancelling, and all of the none-multiples of p cancelling to leave

$$b \equiv \frac{(-1)^{q(p^{m-1}-1)} \cdot ((p-1)p^{m-1}) \cdot ((p-2)p^{m-1}) \cdot \cdots \cdot (p-q)p^{m-1}}{p^{m-1} \cdot 2p^{m-1} \cdot \cdots \cdot qp^{m-1}}.$$

We can now cancel all the powers of p^{m-1} to leave

$$b \equiv \frac{(-1)^{q(p^{m-1}-1)} \cdot (p-1) \cdot (p-2) \cdot \cdots \cdot (p-q)}{1 \cdot 2 \cdot \cdots \cdot q}$$

which we can reevaluate modulo p to give

$$\begin{aligned} b &\equiv (-1)^{q(p^{m-1}-1)+q} \\ &= (-1)^{qp^{m-1}} \\ &= (-1)^{qp}. \end{aligned} \tag{7.3}$$

This concludes the first proof.

The algebraic proof considers the ring over two variables $\mathbb{Z}_p[X, Y]$, which is an integral domain since $\mathbb{Z}_p$ is.

Let

$$a = X^{p^{m-1}} \text{ and } b = Y^{p^{m-1}}. \tag{7.4}$$

Now

$$(a+b)^{p-1}(a+b) = a^p + b^p$$

and

$$(a^{p-1} - a^{p-2}b + a^{p-3}b^2 - \ldots + (-1)^{p-1}b^{p-1})(a+b) = a^p + b^p.$$

Since $\mathbb{Z}_p[X,Y]$ is an integral domain,

$$(a+b)^{p-1} = a^{p-1} - a^{p-2}b + a^{p-3}b^2 - \ldots + (-1)^{p-1}b^{p-1}.$$

Now we apply (7.4) and use the binomial theorem;

$$\begin{aligned}\left(X^{p^{m-1}} + Y^{p^{m-1}}\right)^{p-1} &= (X+Y)^{p^{m-1}(p-1)} \\ &= \sum_{j=0}^{p^{m-1}(p-1)} \binom{p^{m-1}(p-1)}{j} X^{(p^{m-1}(p-1)-j)}Y^j \\ &= (X^{p^{m-1}})^{p-1} - (X^{p^{m-1}})^{p-2}Y^{p^{m-1}} + \ldots \\ &\quad + (-1)^{p-1}(Y^{p^{m-1}})^{p-1}.\end{aligned}$$

Equating coefficients gives the lemma.

□

Results similar to, though not including, the above lemma can be found in [13, 11, 4, 9, 2].

We are now in a position to start our proof of when the zero sequence limit is attained. The first result we should note concerns the $(p-1)p^{m-1}$-th iteration of $U_{p^m,0}$:

LEMMA 7.3: If A is an integer sequence with period p^m, then if $k = (p-1)p^{m-1}$, the sequence $\Delta^k_{(p)}A$, which is the k-th iteration of taking the forward difference of A with each term evaluated modulo p, is periodic with period p^{m-1}.

PROOF: Since A is a linear combination of $U_{p^m,i}$ $(i = 0, 1, \ldots, p^m - 1)$, and Lemma 7.1 tells us that each of the $\Delta^k_{(p)} U_{p^m,i}$ is a translation of $\Delta^k_{(p)} U_{p^m,0}$ we need only show that $\Delta^k_{(p)} U_{p^m,0}$ is p^{m-1}-periodic.

We can see that

$$U_{p^m,0} = (\ 1, \overbrace{0, \ldots, 0}^{n-1}, 0, 0, 0, 1, 0, \ldots, 0, 0, 0, 0, \ldots)$$

$$\Delta^1 U_{p^m,0} = (\ -1, \overbrace{0, \ldots, 0}^{n-2}, 0, 0, 1, -1, 0, \ldots, 0, 0, 0, 1, \ldots)$$

$$\Delta^2 U_{p^m,0} = (\ 1, \overbrace{0, \ldots, 0}^{n-3}, 0, 1, -2, 1, 0, \ldots, 0, 0, 1, -2, \ldots)$$

$$\Delta^3 U_{p^m,0} = (\ -1, \overbrace{0, \ldots, 0}^{n-4}, 1, -3, 3, -1, 0, \ldots, 0, 1, -3, 3, \ldots)$$

$$\vdots$$

$$\Delta^k_{(p^r)} U_{p^m,0} = \left((-1)^k \tbinom{k}{k}, \overbrace{0, \ldots, 0}^{p^{m-1}-1}, \tbinom{k}{0}, -\tbinom{k}{1}, \ldots, (-1)^{k-1}\tbinom{k}{k-1}, (-1)^k \tbinom{k}{k}, \ldots\right)$$
$$= (a_j)_j$$

where

$$a_j = \begin{cases} a_{p^m} & \text{if } j = 0, \\ 0 & \text{if } 0 < j < p^{m-1}, \\ (-1)^{(j+k)} \binom{k}{j-p^{m-1}} & \text{if } p^{m-1} \leqslant j \leqslant p^m, \\ a_{j-p^m} & \text{if } j > p^m. \end{cases}$$

From Lemma 7.2 we know that

$$\binom{k}{\ell} \equiv \begin{cases} (-1)^{qp} \bmod p & \text{if } \ell = qp^{m-1} \quad (0 \leqslant q < p), \\ 0 \bmod p & \text{otherwise.} \end{cases}$$

and hence we find that the terms a_j in $(a_j)_j = \Delta^k U_{p^m,0}$ evaluated modulo p are such that

$$a_j \equiv \begin{cases} (-1)^{(j+k+qp)} \equiv 1 \bmod p & \text{if } j = p^{m-1} + qp^{m-1} \quad (0 \leqslant q < p), \\ 0 \bmod p & \text{otherwise.} \end{cases}$$

Hence, evaluating each term modulo p in $\Delta^k U_{p^m,0}$ we get

$$\Delta^k_{(p)} U_{p^m,0} = (1, \overbrace{0, 0, \ldots, 0}^{p^{m-1}-1}, 1, \overbrace{0, 0, \ldots, 0}^{p^{m-1}-1}, 1, \ldots). \tag{7.5}$$

Hence the lemma.

□

With Lemma 7.3 we can continue to prove the main result of this section.

THEOREM 7.4: If $k = p^m + (r-1)p^{m-1}(p-1)$, and A is an integer sequence with period p^m, then the k-th forward difference of A is congruent to the zero-sequence modulo p^r.

i.e.

$$\Delta^k_{(p^r)} A = (0, 0, 0, \ldots) \tag{7.6}$$

PROOF: This proof is achieved by means of a double inductive argument. We will give a brief overview first to clarify the steps in the proof. We may also wish to refer to Figures 7.1 and 7.2 as examples.

We establish that the p^m-th iteration of the forward difference operator acting on a unit p^m-periodic sequence has each term congruent to 0 modulo p. The $2p^m$-th iteration is then a (linear) combination of this iteration with itself giving a sequence that is zero modulo p^2 (this type of combination of a p^m-periodic sequence with itself is known as a *circular* or *wrapped convolution* [5]. However, using Lemma 7.3 we can prove a tighter result than this.

We can combine the p^m-th iteration with the $p^{m-1}(p-1)$-th iteration (which consists of the sum of a p^{m-1}-periodic sequence and other terms congruent to 0 modulo p) to form the $(p^m + p^{m-1}(p-1))$-th iteration. The terms congruent to 0 modulo p in the $p^{m-1}(p-1)$-th iteration will disappear modulo p^2 in the resulting combination, whilst the p^{m-1}-periodic sequence should have disappeared modulo p^2 by the $p^{m-1} + p^{m-2}(p-1)$-th iteration on from $p^{m-1}(p-1)$ (assuming our result is true for p^{m-1}-periodic sequences) which is before we reach the $(p^m + p^{m-1}(p-1))$-th iteration as required.

Hence the $(p^m + p^{m-1}(p-1))$-th iteration is the zero sequence modulo p^2. We will apply this argument inductively to increase the power of p for which the iterated sequence is congruent to the zero sequence.

The proof in full:

Any n-periodic integer sequence A is a linear combination of the unit n-periodic

sequences $U_{n,i}$. *i.e.* if $A = (a_j)_j$, then

$$A = \sum_{i=0}^{n-1} a_i U_{n,i}.$$

Hence,

$$\Delta^k_{(p^r)} A = \sum_{i=0}^{n-1} a_i \Delta^k_{(p^r)} U_{n,i}.$$

From Lemma 7.1 we know that each of the terms in $\Delta^k_{(p^r)} U_{p^m,i}$ are simply a 'translation' of the terms in $\Delta^k_{(p^r)} U_{p^m,0}$, thus it remains only to show that

$$\Delta^k_{(p^r)} U_{p^m,0} = (0, 0, 0, \ldots)$$

for $k = p^m + (r-1)p^{m-1}(p-1)$.

Continuing the iterative process, seen in the proof of Lemma 7.3, of taking a forward difference of the sequence $U_{p^m,0}$ to the $(p^m - 1)$-th iteration we get, now putting $k = p^m - 1$,

$$\Delta^k U_{p^m,0} = ((-1)^k \tbinom{k}{k}, \tbinom{k}{0}, -\tbinom{k}{1}, \tbinom{k}{2}, \ldots (-1)^{k-1} \tbinom{k}{k-1}, (-1)^k \tbinom{k}{k} \tbinom{k}{k}, \ldots).$$

If $p = 2$, then the first term of $\Delta^k U_{p^m,0}$ is $+1$ and equal to the second term. If $p \neq 2$ and hence odd, then the first term is -1 and the negative of the second term. Thus, the next forward difference is

$$\Delta^{p^m} U_{p^m,0} = (1 + (-1)^{p^m}, \tbinom{p^m}{1}, -\tbinom{p^m}{2}, \tbinom{p^m}{3}, \ldots (-1)^{p^m-1} \tbinom{p^m}{p^m-1}, 1 + (-1)^{p^m}, \ldots). \tag{7.7}$$

In (7.7) the first term is 2 if $p = 2$, and 0 otherwise. The other $p^m - 1$ different terms are $\pm\binom{p^m}{\ell}$ for some $1 \leqslant \ell \leqslant p^m - 1$. Now

$$\binom{p^m}{\ell} = \frac{p^m(p^m-1)(p^m-2)\cdots(p^m-\ell+1)}{1 \cdot 2 \cdot 3 \cdots \ell}$$

and following the same routine as in Lemma 7.2 we can see that for $1 \leqslant \ell \leqslant p^m - 1$,

$$\binom{p^m}{\ell} \equiv 0 \bmod p. \tag{7.8}$$

Hence, all terms in (7.7) evaluate to 0 modulo p. If $(a_j)_j = \Delta^{p^m} U_{p^m,0}$, then the terms $a_1, \ldots, a_{p^m-1}$ are 0 modulo p because they are either plus or minus a binomial coefficient seen in (7.8). a_0 evaluates to 0 modulo p since in the case when p is odd, $a_0 = 1 + (-1)^{p^m} = 0$, and in the case when $p = 2$, $a_0 = 1 + (-1)^{2^m} = 2$.

Hence (7.6) is true for all p^m and $r = 1$.

Putting $m = 1$, and using Lemma 7.3, we see that

$$\Delta_{(p)}^{p-1} U_{p,0} = (1, 1, 1, \ldots)$$

i.e.

$$\Delta^{p-1} U_{p,0} = (1, 1, 1, \ldots) + pA \tag{7.9}$$

for some p-periodic integer sequence A.

Now let us assume that (7.6) is true for $m = 1$ and any $1 \leqslant r \leqslant r'$. Then

$$\Delta_{(p^r)}^{p+(r-1)(p-1)} U_{p,0} = (0, 0, 0, \ldots)\,.$$

Since this is true for $U_{p,0}$ it is true for any p-periodic integer sequence, and in particular,

$$\Delta_{(p^r)}^{p+(r-1)(p-1)} A = (0, 0, 0, \ldots) \qquad \text{for } 1 \leqslant r \leqslant r'$$

i.e.

$$\Delta^{p+(r-1)(p-1)}A = p^r A' \tag{7.10}$$

We also know that $\Delta(1,1,1,\ldots) = (0)$ and so putting $V = \Delta^{p-1}U_{p,0}$,

$$\begin{aligned}\Delta^{p+r(p-1)}U_{p,0} &= \Delta^{p+(r-1)(p-1)}V \\ &= \Delta^{p+(r-1)(p-1)}U_{1,0} + \Delta^{p+(r-1)(p-1)}pA \\ &= (0) + p\Delta^{p+(r-1)(p-1)}A \\ &= (0) + p \cdot p^r A' \\ &= p^{r+1}A'\end{aligned}$$

i.e.

$$\Delta^{p+r(p-1)}_{(p^{r+1})}U_{p,0} = (0,0,0,\ldots) \quad \text{for } 2 \leqslant r+1 \leqslant r'+1 \tag{7.11}$$

and since we know this to be true for $r = 1$,

$$\Delta^{p+(r-1)(p-1)}_{(p^r)}U_{p,0} = (0,0,0,\ldots) \quad \text{for } 1 \leqslant r \leqslant r'+1. \tag{7.12}$$

Hence, by induction, (7.6) is true for $m = 1$ and all $r \in \mathbb{N}$.

Now if we assume that (7.6) is true for $m = m'$, then putting $q = (p-1)p^{m'-1}$ and $k = p^m + (r-1)q$

$$\Delta^{k}_{(p^r)}U_{p^{m'},0} = (0) \quad \text{for } r \in \mathbb{N}. \tag{7.13}$$

Now assuming (7.6) true for $m = m'+1$ and $r \leqslant r'$, then putting $q' = (p-1)p^{m-1}$ and $k' = p^m + (r-1)q'$

$$\Delta^{k'}_{(p^r)}U_{p^m,0} = (0,0,0,\ldots)$$

i.e.

$$\Delta^{k'} U_{p^m,0} = p^r C \quad \text{for } 1 \leqslant r \leqslant r' \tag{7.14}$$

for some p^m-periodic integer sequence C. This means that

$$\begin{aligned} \Delta^{k'+q'} U_{p^m,0} &= \Delta^{k'} \Delta^{q'} U_{p^m,0} \\ &= \Delta^{k'} \left[U_{p^{m-1},0} + pD \right] \\ &= \Delta^{k'} U_{p^{m-1},0} + p\Delta^{k'} D \end{aligned} \tag{7.15}$$

for some p^m-periodic integer sequence D, by (7.5). Since D is a linear combination of the $U_{p^m,i}$ $(i = 0, 1, \ldots, p^m - 1)$, then by (7.14)

$$\Delta^{k'}_{(p^r)} D = (0) \tag{7.16}$$

and since $m - 1 = m'$, and

$$\begin{aligned} k' - p^{m'} + r(p-1)p^{m'-1} &= p^m + (r-1)(p-1)p^{m-1} - p^{m'} + r(p-1)p^{m'-1} \\ &= p^m + rp^m - p^m - rp^{m-1} + p^{m-1} \\ &\quad - (p^{m-1} + rp^{m-1} - rp^{m-2}) \\ &= r\left((p-2)p^{m-1} + p^{m-2}\right) \\ &> 0 \end{aligned}$$

$\Rightarrow$

$$k' > p^{m'} + r(p-1)p^{m'-1},$$

then by (7.13)

$$\Delta^{k'}_{(p^{r+1})} U_{p^{m-1},0} = (0)\,. \tag{7.17}$$

Hence by equations (7.15), (7.16) and (7.17)

$$\Delta^{k'+q'}_{(p^{r+1})} U_{p^m,0} = (0)\,,$$

hence (7.6) is true for all $m, r \in \mathbb{N}$.

□

Examples

See Figures 7.1 and 7.2

Figure 7.1 $U_{2^4,0}$ and its iterated forward differences modulo 2^4.

$$\Delta^0_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ldots)$$
$$\Delta^1_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ldots)$$
$$\Delta^2_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 14,\ldots)$$
$$\Delta^3_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 13,\ 3,\ldots)$$
$$\Delta^4_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 12,\ 6,\ 12,\ldots)$$
$$\Delta^5_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 11,\ 10,\ 6,\ 5,\ldots)$$
$$\Delta^6_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 10,\ 15,\ 12,\ 15,\ 10,\ldots)$$
$$\Delta^7_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 9,\ 5,\ 13,\ 3,\ 11,\ 7,\ldots)$$
$$\Delta^8_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 8,\ 12,\ 8,\ 6,\ 8,\ 12,\ 8,\ldots)$$
$$\Delta^9_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 7,\ 4,\ 12,\ 14,\ 2,\ 4,\ 12,\ 9,\ldots)$$
$$\Delta^{10}_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 6,\ 13,\ 8,\ 2,\ 4,\ 2,\ 8,\ 13,\ 6,\ldots)$$
$$\Delta^{11}_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 0,\ 0,\ 1,\ 5,\ 7,\ 11,\ 10,\ 2,\ 14,\ 6,\ 5,\ 9,\ 11,\ldots)$$
$$\Delta^{12}_{(16)}U_{16,0} = (\ 1,\ 0,\ 0,\ 0,\ 1,\ 4,\ 2,\ 4,\ 15,\ 8,\ 12,\ 8,\ 15,\ 4,\ 2,\ 4,\ldots)$$
$$\Delta^{13}_{(16)}U_{16,0} = (\ 15,\ 0,\ 0,\ 1,\ 3,\ 14,\ 2,\ 11,\ 9,\ 4,\ 12,\ 7,\ 5,\ 14,\ 2,\ 13,\ldots)$$
$$\Delta^{14}_{(16)}U_{16,0} = (\ 1,\ 0,\ 1,\ 2,\ 11,\ 4,\ 9,\ 14,\ 11,\ 8,\ 11,\ 14,\ 9,\ 4,\ 11,\ 2,\ldots)$$
$$\Delta^{15}_{(16)}U_{16,0} = (\ 15,\ 1,\ 1,\ 9,\ 9,\ 5,\ 5,\ 13,\ 13,\ 3,\ 3,\ 11,\ 11,\ 7,\ 7,\ 15,\ldots)$$
$$\Delta^{16}_{(16)}U_{16,0} = (\ 2,\ 0,\ 8,\ 0,\ 12,\ 0,\ 8,\ 0,\ 6,\ 0,\ 8,\ 0,\ 12,\ 0,\ 8,\ 0,\ldots)$$
$$\Delta^{17}_{(16)}U_{16,0} = (\ 14,\ 8,\ 8,\ 12,\ 4,\ 8,\ 8,\ 6,\ 10,\ 8,\ 8,\ 12,\ 4,\ 8,\ 8,\ 2,\ldots)$$
$$\Delta^{18}_{(16)}U_{16,0} = (\ 10,\ 0,\ 4,\ 8,\ 4,\ 0,\ 14,\ 4,\ 14,\ 0,\ 4,\ 8,\ 4,\ 0,\ 10,\ 12,\ldots)$$
$$\Delta^{19}_{(16)}U_{16,0} = (\ 6,\ 4,\ 4,\ 12,\ 12,\ 14,\ 6,\ 10,\ 2,\ 4,\ 4,\ 12,\ 12,\ 10,\ 2,\ 14,\ldots)$$
$$\Delta^{20}_{(16)}U_{16,0} = (\ 14,\ 0,\ 8,\ 0,\ 2,\ 8,\ 4,\ 8,\ 2,\ 0,\ 8,\ 0,\ 14,\ 8,\ 12,\ 8,\ldots)$$
$$\Delta^{21}_{(16)}U_{16,0} = (\ 2,\ 8,\ 8,\ 2,\ 6,\ 12,\ 4,\ 10,\ 14,\ 8,\ 8,\ 14,\ 10,\ 4,\ 12,\ 6,\ldots)$$
$$\Delta^{22}_{(16)}U_{16,0} = (\ 6,\ 0,\ 10,\ 4,\ 6,\ 8,\ 6,\ 4,\ 10,\ 0,\ 6,\ 12,\ 10,\ 8,\ 10,\ 12,\ldots)$$
$$\Delta^{23}_{(16)}U_{16,0} = (\ 10,\ 10,\ 10,\ 2,\ 2,\ 14,\ 14,\ 6,\ 6,\ 6,\ 6,\ 14,\ 14,\ 2,\ 2,\ 10,\ldots)$$
$$\Delta^{24}_{(16)}U_{16,0} = (\ 0,\ 0,\ 8,\ 0,\ 12,\ 0,\ 8,\ 0,\ 0,\ 0,\ 8,\ 0,\ 4,\ 0,\ 8,\ 0,\ldots)$$
$$\Delta^{25}_{(16)}U_{16,0} = (\ 0,\ 8,\ 8,\ 12,\ 4,\ 8,\ 8,\ 0,\ 0,\ 8,\ 8,\ 4,\ 12,\ 8,\ 8,\ 0,\ldots)$$
$$\Delta^{26}_{(16)}U_{16,0} = (\ 8,\ 0,\ 4,\ 8,\ 4,\ 0,\ 8,\ 0,\ 8,\ 0,\ 12,\ 8,\ 12,\ 0,\ 8,\ 0,\ldots)$$
$$\Delta^{27}_{(16)}U_{16,0} = (\ 8,\ 4,\ 4,\ 12,\ 12,\ 8,\ 8,\ 8,\ 8,\ 12,\ 12,\ 4,\ 4,\ 8,\ 8,\ 8,\ldots)$$
$$\Delta^{28}_{(16)}U_{16,0} = (\ 12,\ 0,\ 8,\ 0,\ 12,\ 0,\ 0,\ 0,\ 4,\ 0,\ 8,\ 0,\ 4,\ 0,\ 0,\ 0,\ldots)$$
$$\Delta^{29}_{(16)}U_{16,0} = (\ 4,\ 8,\ 8,\ 12,\ 4,\ 0,\ 0,\ 4,\ 12,\ 8,\ 8,\ 4,\ 12,\ 0,\ 0,\ 12,\ldots)$$
$$\Delta^{30}_{(16)}U_{16,0} = (\ 4,\ 0,\ 4,\ 8,\ 12,\ 0,\ 4,\ 8,\ 12,\ 0,\ 12,\ 8,\ 4,\ 0,\ 12,\ 8,\ldots)$$

$\Delta^{31}_{(16)}U_{16,0} = (\ 12,\ 4,\ 4,\ 4,\ 4,\ 4,\ 4,\ 4,\ 4, 12, 12, 12, 12, 12, 12, 12, \ldots)$

$\Delta^{32}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{33}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 8,\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 8, \ldots)$

$\Delta^{34}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 8,\ 0,\ 8,\ 0,\ 0,\ 0,\ 0,\ 0,\ 8,\ 0, \ldots)$

$\Delta^{35}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 0,\ 0,\ 8,\ 8,\ 8,\ 8,\ 0,\ 0,\ 0,\ 0,\ 8,\ 8,\ 8, \ldots)$

$\Delta^{36}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 0,\ 8,\ 0,\ 0,\ 0,\ 8,\ 0,\ 0,\ 0,\ 8,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{37}_{(16)}U_{16,0} = (\ 8,\ 0,\ 0,\ 8,\ 8,\ 0,\ 0,\ 8,\ 8,\ 0,\ 0,\ 8,\ 8,\ 0,\ 0,\ 8, \ldots)$

$\Delta^{38}_{(16)}U_{16,0} = (\ 8,\ 0,\ 8,\ 0,\ 8,\ 0,\ 8,\ 0,\ 8,\ 0,\ 8,\ 0,\ 8,\ 0,\ 8,\ 0, \ldots)$

$\Delta^{39}_{(16)}U_{16,0} = (\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8,\ 8, \ldots)$

$\Delta^{40}_{(16)}U_{16,0} = (\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

Figure 7.2 $U_{3^3,0}$ and its iterated forward differences modulo 3^3.

$\Delta^{0}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ \ldots)$

$\Delta^{1}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ \ldots)$

$\Delta^{2}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 25,\ \ldots)$

$\Delta^{3}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 24,\ 3,\ \ldots)$

$\Delta^{4}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 23,\ 6,\ 23,\ \ldots)$

$\Delta^{5}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 22,\ 10,\ 17,\ 5,\ \ldots)$

$\Delta^{6}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 21,\ 15,\ 7,\ 15,\ 21,\ \ldots)$

$\Delta^{7}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 20,\ 21,\ 19,\ 8,\ 6,\ 7,\ \ldots)$

$\Delta^{8}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 19,\ 1,\ 25,\ 16,\ 25,\ 1,\ 19,\ \ldots)$

$\Delta^{9}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 18,\ 9,\ 24,\ 18,\ 9,\ 3,\ 18,\ 9,\ \ldots)$

$\Delta^{10}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 17,\ 18,\ 15,\ 21,\ 18,\ 21,\ 15,\ 18,\ 17,\ \ldots)$

$\Delta^{11}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 16,\ 1,\ 24,\ 6,\ 24,\ 3,\ 21,\ 3,\ 26,\ 11,\ \ldots)$

$\Delta^{12}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 15,\ 12,\ 23,\ 9,\ 18,\ 6,\ 18,\ 9,\ 23,\ 12,\ 15,\ \ldots)$

$\Delta^{13}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 14,\ 24,\ 11,\ 13,\ 9,\ 15,\ 12,\ 18,\ 14,\ 16,\ 3,\ 13,\ \ldots)$

$\Delta^{14}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 13,\ 10,\ 14,\ 2,\ 23,\ 6,\ 24,\ 6,\ 23,\ 2,\ 14,\ 10,\ 13,\ \ldots)$

$\Delta^{15}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 12,\ 24,\ 4,\ 15,\ 21,\ 10,\ 18,\ 9,\ 17,\ 6,\ 12,\ 23,\ 3,\ 15,\ \ldots)$

$\Delta^{16}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 11,\ 12,\ 7,\ 11,\ 6,\ 16,\ 8,\ 18,\ 8,\ 16,\ 6,\ 11,\ 7,\ 12,\ 11,\ \ldots)$

$\Delta^{17}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 10,\ 1,\ 22,\ 4,\ 22,\ 10,\ 19,\ 10,\ 17,\ 8,\ 17,\ 5,\ 23,\ 5,\ 26,\ 17,\ \ldots)$

$\Delta^{18}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 9,\ 18,\ 21,\ 9,\ 18,\ 15,\ 9,\ 18,\ 7,\ 18,\ 9,\ 15,\ 18,\ 9,\ 21,\ 18,\ 9,\ \ldots)$

$\Delta^{19}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 8,\ 9,\ 3,\ 15,\ 9,\ 24,\ 21,\ 9,\ 16,\ 11,\ 18,\ 6,\ 3,\ 18,\ 12,\ 24,\ 18,\ 19,\ \ldots)$

$\Delta^{20}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 7,\ 1,\ 21,\ 12,\ 21,\ 15,\ 24,\ 15,\ 7,\ 22,\ 7,\ 15,\ 24,\ 15,\ 21,\ 12,\ 21,\ 1,\ 7,\ \ldots)$

$\Delta^{21}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 6,\ 21,\ 20,\ 18,\ 9,\ 21,\ 9,\ 18,\ 19,\ 15,\ 12,\ 8,\ 9,\ 18,\ 6,\ 18,\ 9,\ 7,\ 6,\ 21,\ \ldots)$

$\Delta^{22}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 0,\ 0,\ 1,\ 5,\ 15,\ 26,\ 25,\ 18,\ 12,\ 15,\ 9,\ 1,\ 23,\ 24,\ 23,\ 1,\ 9,\ 15,\ 12,\ 18,\ 25,\ 26,\ 15,\ 5,\ \ldots)$

$\Delta^{23}_{(27)}U_{27,0} = (\ 26,\ 0,\ 0,\ 0,\ 1,\ 4,\ 10,\ 11,\ 26,\ 20,\ 21,\ 3,\ 21,\ 19,\ 22,\ 1,\ 26,\ 5,\ 8,\ 6,\ 24,\ 6,\ 7,\ 1,\ 16,\ 17,\ 23,\ \ldots)$

$\Delta^{24}_{(27)}U_{27,0} = (\ 1,\ 0,\ 0,\ 1,\ 3,\ 6,\ 1,\ 15,\ 21,\ 1,\ 9,\ 18,\ 25,\ 3,\ 6,\ 25,\ 6,\ 3,\ 25,\ 18,\ 9,\ 1,\ 21,\ 15,\ 1,\ 6,\ 3,\ \ldots)$

$\Delta^{25}_{(27)}U_{27,0} = (\ 26,\ 0,\ 1,\ 2,\ 3,\ 22,\ 14,\ 6,\ 7,\ 8,\ 9,\ 7,\ 5,\ 3,\ 19,\ 8,\ 24,\ 22,\ 20,\ 18,\ 19,\ 20,\ 21,\ 13,\ 5,\ 24,\ 25,\ \ldots)$

$\Delta^{26}_{(27)}U_{27,0} = (\ 1,\ 1,\ 1,\ 1,\ 19,\ 19,\ 19,\ 1,\ 1,\ 1,\ 25,\ 25,\ 25,\ 16,\ 16,\ 16,\ 25,\ 25,\ 25,\ 1,\ 1,\ 1,\ 19,\ 19,\ 19,\ 1,\ 1,\ \ldots)$

$\Delta^{27}_{(27)}U_{27,0} = (\ 0,\ 0,\ 0,\ 18,\ 0,\ 0,\ 9,\ 0,\ 0,\ 24,\ 0,\ 0,\ 18,\ 0,\ 0,\ 9,\ 0,\ 0,\ 3,\ 0,\ 0,\ 18,\ 0,\ 0,\ 9,\ 0,\ 0,\ \ldots)$

$\Delta^{28}_{(27)}U_{27,0} = (\ 0,\ 0,\ 18,\ 9,\ 0,\ 9,\ 18,\ 0,\ 24,\ 3,\ 0,\ 18,\ 9,\ 0,\ 9,\ 18,\ 0,\ 3,\ 24,\ 0,\ 18,\ 9,\ 0,\ 9,\ 18,\ 0,\ 0,\ \ldots)$

$\Delta^{29}_{(27)}U_{27,0} = (\ 0,\ 18,\ 18,\ 18,\ 9,\ 9,\ 9,\ 24,\ 6,\ 24,\ 18,\ 18,\ 18,\ 9,\ 9,\ 9,\ 3,\ 21,\ 3,\ 18,\ 18,\ 18,\ 9,\ 9,\ 9,\ 0,\ 0,\ \ldots)$

$\Delta^{30}_{(27)}U_{27,0} = (\ 18,\ 0,\ 0,\ 18,\ 0,\ 0,\ 15,\ 9,\ 18,\ 21,\ 0,\ 0,\ 18,\ 0,\ 0,\ 21,\ 18,\ 9,\ 15,\ 0,\ 0,\ 18,\ 0,\ 0,\ 18,\ 0,\ 0,\ \ldots)$

$\Delta^{31}_{(27)}U_{27,0} = (\ 9,\ 0, 18,\ 9,\ 0, 15, 21,\ 9,\ 3,\ 6,\ 0, 18,\ 9,\ 0, 21, 24, 18,\ 6, 12,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0, 18, \ldots)$

$\Delta^{32}_{(27)}U_{27,0} = (\ 18, 18, 18, 18, 15,\ 6, 15, 21,\ 3, 21, 18, 18, 18, 21,\ 3, 21, 15,\ 6, 15, 18, 18, 18, 18, 18, 18, 18, 18, \ldots)$

$\Delta^{33}_{(27)}U_{27,0} = (\ 0,\ 0,\ 0, 24, 18,\ 9,\ 6,\ 9, 18, 24,\ 0,\ 0,\ 3,\ 9, 18, 21, 18,\ 9,\ 3,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{34}_{(27)}U_{27,0} = (\ 0,\ 0, 24, 21, 18, 24,\ 3,\ 9,\ 6,\ 3,\ 0,\ 3,\ 6,\ 9,\ 3, 24, 18, 21, 24,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{35}_{(27)}U_{27,0} = (\ 0, 24, 24, 24,\ 6,\ 6,\ 6, 24, 24, 24,\ 3,\ 3,\ 3, 21, 21, 21,\ 3,\ 3,\ 3,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{36}_{(27)}U_{27,0} = (\ 24,\ 0,\ 0,\ 9,\ 0,\ 0, 18,\ 0,\ 0,\ 6,\ 0,\ 0, 18,\ 0,\ 0,\ 9,\ 0,\ 0, 24,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{37}_{(27)}U_{27,0} = (\ 3,\ 0,\ 9, 18,\ 0, 18,\ 9,\ 0,\ 6, 21,\ 0, 18,\ 9,\ 0,\ 9, 18,\ 0, 24,\ 3,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, 24, \ldots)$

$\Delta^{38}_{(27)}U_{27,0} = (\ 24,\ 9,\ 9,\ 9, 18, 18, 18,\ 6, 15,\ 6, 18, 18, 18,\ 9,\ 9,\ 9, 24,\ 6, 24,\ 0,\ 0,\ 0,\ 0,\ 0, 24,\ 6, \ldots)$

$\Delta^{39}_{(27)}U_{27,0} = (\ 12,\ 0,\ 0,\ 9,\ 0,\ 0, 15,\ 9, 18, 12,\ 0,\ 0, 18,\ 0,\ 0, 15,\ 9, 18,\ 3,\ 0,\ 0,\ 0,\ 0,\ 0, 24,\ 9, 18, \ldots)$

$\Delta^{40}_{(27)}U_{27,0} = (\ 15,\ 0,\ 9, 18,\ 0, 15, 21,\ 9, 21, 15,\ 0, 18,\ 9,\ 0, 15, 21,\ 9, 12, 24,\ 0,\ 0,\ 0,\ 0, 24, 12,\ 9, 21, \ldots)$

$\Delta^{41}_{(27)}U_{27,0} = (\ 12,\ 9,\ 9,\ 9, 15,\ 6, 15, 12, 21, 12, 18, 18, 18, 15,\ 6, 15,\ 3, 12,\ 3,\ 0,\ 0,\ 0, 24, 15, 24, 12, 21, \ldots)$

$\Delta^{42}_{(27)}U_{27,0} = (\ 24,\ 0,\ 0,\ 6, 18,\ 9, 24,\ 9, 18,\ 6,\ 0,\ 0, 24, 18,\ 9, 15,\ 9, 18, 24,\ 0,\ 0, 24, 18,\ 9, 15,\ 9, 18, \ldots)$

$\Delta^{43}_{(27)}U_{27,0} = (\ 3,\ 0,\ 6, 12, 18, 15, 12,\ 9, 15, 21,\ 0, 24, 21, 18,\ 6, 21,\ 9,\ 6,\ 3,\ 0, 24, 21, 18,\ 6, 21,\ 9,\ 6, \ldots)$

$\Delta^{44}_{(27)}U_{27,0} = (\ 24,\ 6,\ 6,\ 6, 24, 24, 24,\ 6,\ 6,\ 6, 24, 24, 24, 15, 15, 15, 24, 24, 24, 24, 24, 24, 15, 15, 15, 24, 24, \ldots)$

$\Delta^{45}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0, 18,\ 0,\ 0,\ 9,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0, 18,\ 0,\ 0,\ 9,\ 0,\ 0, \ldots)$

$\Delta^{46}_{(27)}U_{27,0} = (\ 18,\ 0, 18,\ 9,\ 0,\ 9, 18,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0,\ 9, 18,\ 0,\ 0,\ 0,\ 0, 18,\ 9,\ 0,\ 9, 18,\ 0,\ 9, \ldots)$

$\Delta^{47}_{(27)}U_{27,0} = (\ 9, 18, 18, 18,\ 9,\ 9,\ 9, 18, 18, 18, 18, 18, 18,\ 9,\ 9,\ 9,\ 0,\ 0,\ 0, 18, 18, 18,\ 9,\ 9,\ 9,\ 9,\ 9, \ldots)$

$\Delta^{48}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0, 18,\ 0,\ 0,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{49}_{(27)}U_{27,0} = (\ 18,\ 0, 18,\ 9,\ 0,\ 9, 18,\ 0,\ 0,\ 0,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0,\ 0,\ 0,\ 0,\ 9, \ldots)$

$\Delta^{50}_{(27)}U_{27,0} = (\ 9, 18, 18, 18,\ 9,\ 9,\ 9,\ 0,\ 0,\ 0, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18,\ 0,\ 0,\ 0,\ 9,\ 9, \ldots)$

$\Delta^{51}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0, \ldots)$

$\Delta^{52}_{(27)}U_{27,0} = (\ 18,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0, 18,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, \ldots)$

$\Delta^{53}_{(27)}U_{27,0} = (\ 9, 18, 18, 18, 18, 18, 18, 18, 18, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9, \ldots)$

$\Delta^{54}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

$\Delta^{55}_{(27)}U_{27,0} = (\ 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9, \ldots)$

$\Delta^{56}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 9, \ldots)$

$\Delta^{57}_{(27)}U_{27,0} = (\ 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0, 18,\ 0,\ 0,\ 0,\ 0,\ 0,\ 9,\ 0,\ 0, \ldots)$

$\Delta^{58}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0, 18,\ 9,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0, 18,\ 9,\ 0,\ 0,\ 0,\ 0,\ 9, 18,\ 0, 18, \ldots)$

$\Delta^{59}_{(27)}U_{27,0} = (\ 18,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9, 18, 18, 18,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9, 18, 18, 18,\ 0,\ 0,\ 0,\ 9,\ 9,\ 9, 18, 18, \ldots)$

$\Delta^{60}_{(27)}U_{27,0} = (\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0,\ 9,\ 0,\ 0, \ldots)$

$\Delta^{61}_{(27)}U_{27,0} = (\ 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, 18,\ 0,\ 9, \ldots)$

$\Delta^{62}_{(27)}U_{27,0} = (\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9,\ 9, \ldots)$

$\Delta^{63}_{(27)}U_{27,0} = (\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0, \ldots)$

7.3 Alternating Sums of Binomial Coefficients

Another way of calculating the values of $\Delta^k U_{p^m,0}$ is in terms of alternating sums of binomial coefficients.

We have seen that the first $p^m - 1$ iterations of Δ on $U_{p^m,0}$ generate terms that are either zero or plus or minus a binomial coefficient. After the $(p^m - 1)$-th iteration, the binomial coefficients cascading down from each periodic value 1 in $U_{p^m,0}$ start to 'interfere' with their adjoining sets of binomial cofficients. This is perhaps best seen in an example (see Figures 7.1 and 7.2).

In fact we can calculate the value of the i-th term of $\Delta^k U_{p^m,0}$ in terms of an alternating sum of binomial coefficients, for $i = 0, 1, \ldots, p^m - 1$, as

$$\Delta^k U_{p^m,0}{}_{j=0}^{p^m-1} = \left(\sum_{\ell=\min(1,j)}^{\left\lfloor \frac{k+j}{p^m} \right\rfloor} (-1)^{k-p^m\ell+j} \binom{k}{p^m\ell - j} \right)_{j=0}^{p^m-1}.$$

This and Theorem 7.4 leads us to the following

COROLLARY 7.5:

$$\sum_{\ell=\min(1,j)}^{\left\lfloor \frac{k+j}{p^m} \right\rfloor} (-1)^{k-p^m\ell+j} \binom{k}{p^m\ell - j} \equiv 0 \bmod p^r$$

for $k = p^m + p^{m-1}(r-1)(p-1)$, any $j \in \{0, 1, 2, \ldots, p^m - 1\}$, any $r, m \in \mathbb{N}$ and p prime.

Examples

We put $p = 3$, $m = 2$, $r = 4$ which means $k = 27$, and we run j from 0 to 8. Since p, m, r and j are variable in the sum, many more examples are available!

$$\sum_{0}^{3}(-1)^{k-3^2\ell+0}\binom{k}{3^2\ell-0} = -1 + 4686825 - 4686825 + 1$$
$$= 0$$
$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+1}\binom{k}{3^2\ell-1} = -2220075 + 8436285 - 27$$
$$= 6216183$$
$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+2}\binom{k}{3^2\ell-2} = 888030 - 13037895 + 351$$
$$= -12149514$$
$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+3}\binom{k}{3^2\ell-3} = -296010 + 17383860 - 2925$$
$$= 17084925$$
$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+4}\binom{k}{3^2\ell-4} = 80730 - 20058300 + 17550$$

$$= -19960020$$

$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+5}\binom{k}{3^2\ell-5} = -17550 + 20058300 - 80730$$

$$= 19960020$$

$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+6}\binom{k}{3^2\ell-6} = 2925 - 17383860 + 296010$$

$$= -17084925$$

$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+7}\binom{k}{3^2\ell-7} = -351 + 13037895 - 888030$$

$$= 12149514$$

$$\equiv 0 \bmod 3^4$$

$$\sum_{1}^{3}(-1)^{k-3^2\ell+8}\binom{k}{3^2\ell-8} = 27 - 8436285 + 2220075$$

$$= -6216183$$

$$\equiv 0 \bmod 3^4$$

Chapter 8

Construction of a PGP

8.1 Polynomials and Sequences

Having found a sequence, A, to produce a certain shape in a PGP, we need to find a polynomial, $f \in \mathbb{Z}[x]$, such that $f(j) \equiv a_j \bmod p^m$ for $j = 0, 1, \ldots, p^m - 1$. Does such a polynomial always exist?

i.e. , for a given $p^m \in \mathbf{P}$ and an integer sequence $A = (a_j)_j$ with period p^m, is there a polynomial $f(x) \in \mathbb{Z}[x]$ with $f(j) \equiv a_j \bmod p^m$?

LEMMA 8.1: For each p^m-periodic sequence (over $\mathbb{Z}$), $(a_j)_j$, there exists a polynomial $f(x) \in \mathbb{Z}[x]$ such that $f(j) \equiv a_j \bmod p^m$ if and only if the polynomial $u_{p^m}(x) \in \mathbb{Z}[x]$ exists, where

$$u_{p^m}(0) \equiv 1 \bmod p^m$$

$$u_{p^m}(i) \equiv 0 \bmod p^m \text{ for } i = 1, 2, \ldots, p^m - 1.$$

PROOF: If $f(x) \in \mathbb{Z}[x]$, then $f(x - i) \in \mathbb{Z}[x]$ $(i \in \mathbb{Z})$, and the sequence produced by $f(j - i) \bmod p^m$ is the same as the sequence generated by $f(j) \bmod p^m$

with each term shifted i places to the right. Given a polynomial $u_{p^m}(x)$ as above, then

$$f(x) = \sum_{j=0}^{p^m-1} a_j u_{p^m}(x-j),$$

hence it is sufficient to find such a polynomial, $u_{p^m}(x)$.

It is necessary since $U_{p^m,0}$ is a p^m-periodic sequence of integers, and hence corresponds to a feasible shape.

□

8.2 Construction of a Feasible Shape by a PGP

We will now attempt to find methods of construction of feasible shapes by means of a PGP. We will show that a feasible shape defined by a p^m-periodic sequence evaluated modulo p^m, $(a_j)_j$, is *not* always possible to construct by means of a PGP. However, the same shape can be constructed by the sequence $(p^r a_j)_j$ evaluated modulo p^{m+r} for some r to be determined.

Iterated Forward Differences

By means of iterating the forward difference operator on polynomials and comparing with the results on iterating the forward difference operator on unit periodic sequences from the previous chapter, we will show that not every p^m-periodic sequence of integers can be generated by an integer-cofficient polynomial. However, we will attempt to get around this problem by showing that a particular shape associated with a p^m-periodic integer sequence evaluated modulo p^m is also associated

with a p^m-periodic sequence evaluated modulo p^{m+r} for some $r \in \mathbb{N}$. Note that the division of the circle into p^{m+r} equal angles in the new construction, instead of the p^m angles in the original, means that each term in the sequence must be multiplied by p^r so that it represents the same angular displacement from the Ox-axis. This means that we are no longer trying to find a polynomial $u_{p^m}(x)$ and instead are trying to find a polynomial $v(x)$ such that

$$v(i) \equiv \begin{cases} p^r \bmod p^{m+r} & \text{if } i \equiv 0 \bmod p^m \\ 0 \bmod p^{m+r} & \text{otherwise.} \end{cases}$$

LEMMA 8.2: The polynomial $u_{p^m}(x) \in \mathbb{Z}[x]$ exists if and only if $m = 1$.

PROOF: The sequence generated by $\bar{u}_{p^m}(x)$, the reduced polynomial given by **reduce**(u, p^m), over the values $x = 0, 1, 2, \ldots$ is the unit periodic sequence $U_{p^m,0} = (1, 0, 0, \ldots, 1, 0, 0, \ldots)$.

It is easy to show that the sequence generated by $\Delta\bar{u}_{p^m}(x)$ over $x = 0, 1, 2, \ldots$ is $\Delta U_{p^m,0}$, and similarly for iterated forward differences.

We know from Theorem 7.4 that the sequence $U_{p^m,0}$ takes exactly $p^{m-1}(pm - m + 1)$ iterations of the forward difference operator before it becomes the zero sequence (modulo p^m). Hence $u_{p^m}(x)$ takes $p^{m-1}(pm - m + 1)$ iterations of the forward difference operator before evaluating to zero modulo p^m.

However, we know from Lemma 5.1 that if $u_{p^m}(x)$ exists, then an equivalent polynomial $\bar{u}_{p^m}(x)$ of degree less than s_{p^m} also exists generating the same sequence and hence its forward differences also generate the same sequences as those of $u_{p^m}(x)$. Since $\bar{u}_{p^m}(x)$ is of degree less than or equal to s_{p^m}, the s_{p^m}-th iteration of the forward difference operator acting on $\bar{u}_{p^m}(x)$ is the zero polynomial, since the forward difference operator reduces the degree of a polynomial by (at least) one.

Now, for $m \geqslant 2$,

$$\mathbf{m}_{p^{m-1}(pm-m+1),p} \geqslant (pm - m + 1)\frac{p^{m-1} - 1}{p - 1}$$

since each p^{m-1} contributes at least $\frac{p^{m-1}-1}{p-1}$ to $\mathbf{m}_{p^{m-1}(pm-m+1),p}$

$$\begin{aligned} &= m(p-1)\frac{p^{m-1} - 1}{p - 1} + \frac{p^{m-1} - 1}{p - 1} \\ &> m(p-1)\frac{p^{m-1} - 1}{p - 1} \\ &\geqslant m \end{aligned}$$

and so $p^{m-1}(pm - m + 1) > \mathbf{s}_{p^m}$.

Hence the sequence $U_{p^m,0}$ cannot be generated by any integer coefficient polynomial evaluated modulo p^m for $m \geqslant 2$.

When $m = 1$, we are considering integer coefficient polynomials evaluated over the field $\mathbb{Z}_p$. Since $\mathbb{Z}_p$ is a field, we can find a polynomial in $\mathbb{Z}[x]$ for *any* p-periodic sequence of integers. this can be achieved using typical polynomial interpolation methods since the resulting polynomial will have rational coefficients which exist within $\mathbb{Z}_p$ and hence have an equivalent value in $\mathbb{Z}$ when evaluated modulo p. Hence the sequence $U_{p,0}$ can be generated by a polynomial in $\mathbb{Z}[x]$.

□

8.3 Constructing Polynomials

We can construct a polynomial, f, from a feasible sequence that will take on as many values from the start of the sequence as possible, by adding a certain multiple

of $x(x-1)\cdots(x-j)$ to f as $j = 0, 1, 2, \ldots, \mathbf{s}_{p^m}$. Each of these polynomials will not affect the first j values of f, but will affect those onwards. We control the amount of $x(x-1)\cdots(x-j)$ added on to f to affect the $j+1$-th value to the desired value from the given sequence. We can then continue with $x(x-1)\cdots(x-j+1)$.

A procedure that implements this method is **interp** found in Procedure 8.1.

Procedure 8.1 A procedure that takes a p^m-periodic integer sequence, A, and tries to formulate a polynomial f such that f generates A when evaluated over the integers modulo p^m.

---

procedure **interp** $(A = (a_j)_j \in \mathbb{Z}_{p^m}{}^{p^m}, p^m \in \mathbf{P}) \rightarrow (\mathbb{Z}_{p^m}[x])$

if $\Delta^{\mathbf{s}_{p^m}}_{(p^m)} A = (0)_j$ then

$f(x) := 0$

for $j := 0$ to $\mathbf{s}_{p^m} - 1$

$z(x) := x(x-1)\cdots(x-(j-1))$

if $p | p(a_j - f(j))/z(j)$ then

$f(x) := f(x) + z(x)(a_j - f(j))/z(j)$

else

exit procedure

end if

end for

return **reduce**$(f(x), p^m)$

else

exit procedure

end if

Notes on Procedure 8.1

The test $p|p(a_j - f(j))/z(j)$ is simply to determine that the power of p dividing $a_j - f(j)$ is greater than or equal to the power of p dividing $z(j)$. This is required so that $\frac{a_j - f(j)}{z(j)}$ exists modulo p^m.

Note that if this test fails, then the value $f(j)$ cannot be made to equal a_j modulo p^m without affecting some of the values of $f(i)$, $0 \leqslant i < j$, which are already correct (*i.e.* $f(i) \equiv a_i \bmod p^m$). In this case the procedure exits with no return value.

Similarly, if the input sequence A does not become the zero sequence 'quickly enough', *i.e.* after the largest degree polynomial in $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$ (of degree $\mathrm{s}_{p^m} - 1$) has iterated to the zero polynomial under the same operator, then the procedure exits with no return value.

Example

If we wished to generate the sequence $(1, 5, 3, 7, 5, 1, 7, 3, \ldots)$ by a polynomial evaluated modulo 2^3, the procedure **interp**$((1, 5, 3, 7, 5, 1, 7, 3, \ldots), 2^3)$ gives:

Starting with $j = 0$ and $f(x) = 0$,

$f(x) = f(x) + 1z_{(j-1)}(=)1.$

With $j = 1$

$f(x) = f(x) + 4z_{(j-1)}(=)4x + 1.$

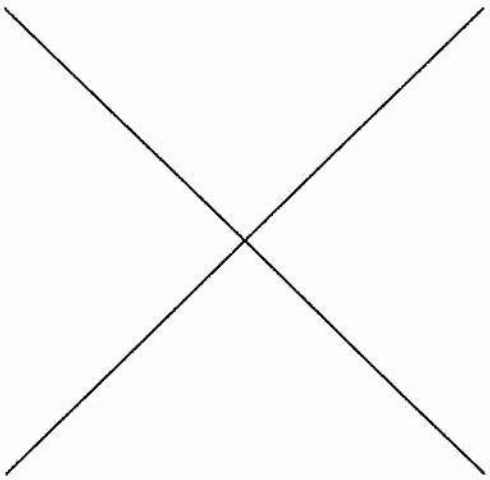
Then with $j = 2$

$f(x) = f(x) + 5z_{(j-1)}(=)5x^2 - x + 1.$

With $j = 3$

$f(x) = f(x) + 2z_{(j-1)}(=)2x^3 - x^2 + 3x + 1,$

and finally, **reduce**$(f, p^m) = 2x^3 + 3x^2 + 7x + 1$



$\mathcal{P}_{2x3+3x2+7x+1,8}$

8.4 Increased Power Constructions

Lemma 8.2 hints at how we might resolve the problem of not being able to generate any feasible shape from a polynomial evaluated modulo p^m.

We should note that the shape generated by the p^m-periodic sequence $A = (a_j)_j$ modulo p^m, is the same shape generated by the p^m-periodic sequence $p^r A = (p^r a_j)_j$ modulo p^{m+r}.

The key here is that $\mathbf{s}_{p^{m+r}}$ increases as r increases, whilst the number of iterations of the forward difference operator needed to reduce $p^r A$ to the zero sequence modulo

p^{m+r} is $p^m+p^{m-1}(p-1)(m-1)$, the same as that of A. This is because the terms in p^rA already have a factor of p^r throughout, and the sequence is still p^m-periodic, not p^{m+r}-periodic as any general sequence might be, when working with PGPs modulo p^{m+r}.

We can determine how large r might be by increasing r until we have $\mathrm{s}_{p^{m+r}} \geqslant p^m+p^{m-1}(p-1)(m-1)$. Unfortunately, for larger values of m the value of r increases rapidly, making computations more difficult if not impractical. However, a modest amount of success can be achieved for some simpler examples.

Example

When creating a PGP to display the symmetries of the most symmetric frieze group (see Figure 1.3) seen at the end of Chapter 6 we first find the sequence and value of p^m associated with the (feasible) shape required, namely

$$A = (0, 2, 0, 6, 4, 6, 0, 2, \ldots)$$

with $p^m = 2^3$.

Procedure **interp** does not attempt an appropriate value of f since the sequence does not iterate to the zero sequence (modulo p^m) quickly enough under the firward difference operator (*i.e.* the sequence takes longer to disappear than the most stubborn polynomial in $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$ of degree s_{p^m}.

In this case, the first iteration of $\Delta^n_{(p^m)}A$ that equals the zero sequence is $n = 8$, and $\mathrm{s}_{2^3} = 4$.

We wish to find r such that $\mathbf{s}_{p^{m+r}} \geqslant 8$, and the smallest such r is 2 (found by incrementing r from 0 until the condition is satisfied).

We are now looking for a polynomial $f \in \mathbb{Z}[x]/\mathcal{Z}_{p^{m+r},x}$ such that

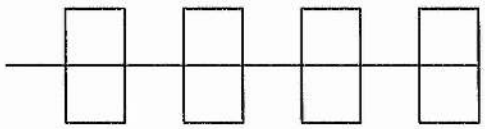
$$\left(f(j) \bmod p^{m+r}\right) = p^r A$$

i.e.

$$\left(f(j) \bmod 2^5\right) = (0, 8, 0, 24, 16, 24, 0, 8, \ldots)$$

Using **interp** now yields

$$f(x) = \textbf{interp}(4A, 2^5) = x^7 + 11x^3 + 8x^2 + 20x$$



$\mathcal{P}_{x^7+11x^3+8x^2+20x,32}$

Chapter 9

Special Cases

9.1 Rotating Quadratics

In this section we will attempt to apply the procedures formulated in Part II to find some interesting closed PGPs generated by quadratics $f(x) = ax^2 + bx + c$ evaluated modulo prime powers, p^m.

$p^m \not| a, b$.

For simplicity, let us assume that the first repeat of our polynomial sequence modulo p^m is p^m.

Rotational Symmetry

For rotational symmetry, we need to find $m' < m$ (where $\rho = p^{m-m'}$ is the degree of rotational symmetry) such that

$$f(x + p^{m'}) - f(x) = c + z(x)$$

for some $z \in \mathcal{Z}_{p^m,x}$. *i.e.*

$$2ap^{m'}x + ap^{2m'} + bp^{m'} = c + z(x)$$

for some non-zero constant $c \in \mathbb{Z}_{p^m}$ and $z \in \mathcal{Z}_{p^m,x}$.

The easiest way to ensure this would be to have

$$p^{m-m'}|a \text{ and } p^{m-m'} \not| b. \tag{9.1}$$

Reflectional Symmetry

The conditions required for reflectional symmetry given in equations (4.13) and (4.16) after some algebra, require either

$$2ax(x+1) \in \mathcal{Z}_{p^m,x} \tag{9.2}$$

or

$$2ax^2 \in \mathcal{Z}_{p^m,x}. \tag{9.3}$$

This can only happen when $p^m|a$, which is a case we are discounting since then f is not a 'true' quadratic, or $p = 2$ and $p^{m-2}|a$ in the case of (9.2) or $p = 2$ and $p^{m-1}|a$ in the case of (9.3). Hence, any quadratic evaluated modulo p^m with $p > 2$ cannot display reflectional symmetry.

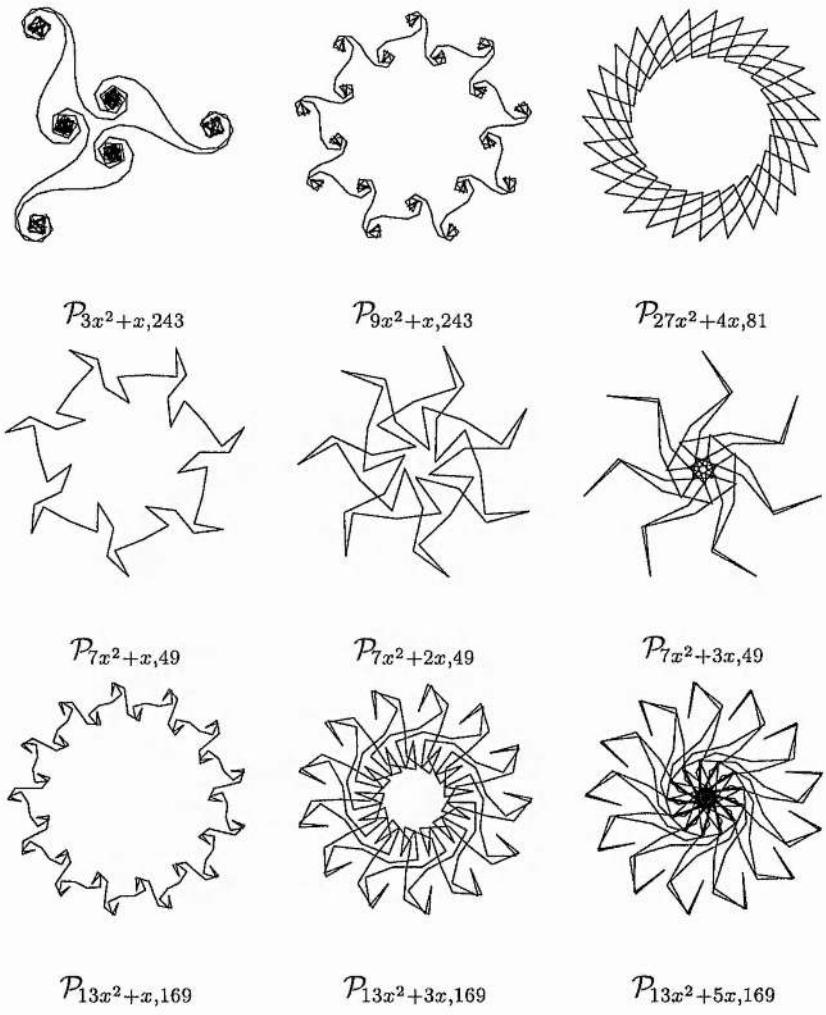
Examples of Constructing Quadratic PGPs of Given Symmetry Group

From (9.1) we can put $a = p^{m-m'}$ and b coprime to p to generate a PGP with rotational symmetry of order $p^{m-m'}$.

We present some examples in Figure 9.1.

Further examples of quadratic generated PGPs can be seen on the cover title page.

Figure 9.1 Some closed PGPs with high degrees of rotational symmetry generated from quadratics.



9.2 Number of Shapes modulo p

In this section we look more closely at the specific cases when $m = 1$ with closed PGPs, *i.e.* we are looking at PGPs with polynomials, $f(x) \in \mathbb{Z}[x]$ evaluated over the integers modulo a prime, p. We will be interested to count the number of different closed shapes, and discard repeats of the same basic shapes, that is shapes that are the same through rotation or reflection. We should note that since we are working over a field $\mathbb{Z}_p$, any p-periodic sequence of integers modulo p can be generated by an integer coefficient polynomial.

Since we are interested in closed PGPs, we should note that a sequence generating a closed PGP modulo a prime p, must span the whole of $\mathbb{Z}_p$, since Procedure 4.1 requires

$$\Phi_p(x) \text{divides} \sum_{j=0}^{p-1} x^{f(j) \bmod p}$$

and the cyclotomic polynomial

$$\Phi_p(x) = 1 + x + x^2 + \ldots + x^{p-1}.$$

Counting

There are $\frac{1}{2}(p-1)!$ different sequences that start with 0, ignoring any reverses of already counted sequences.

$u = \frac{p-1}{2}$ of these are equilateral p-gons (mostly 'star-shaped'), with constant angular differences of $1, 2, \ldots, u$. Each of these sequences/shapes have symmetry group D_p. There is exactly one arrangement of each equilateral p-gon in our original counting of $\frac{1}{2}(p-1)!$.

Figure 9.2 Number of shapes of each symmetry group for PGPs evaluated modulo p.

p	group D_p	group D_1	trivial group	**total**
3	1	0	0	1
5	2	2	0	4
7	3	21	15	39
11	5	1915	81515	83435

We now count those sequences that correspond to shapes with symmetry group D_1, being careful to subtract the u arrangements with symmetry group D_p. These arrangements are *palindromic*, and there are

$$\left(\tfrac{p-1}{2}\right)(p-3)(p-5)\ldots - u = u!2^{u-1} - u$$

different arrangements.

The remaining sequences correspond to shapes with the trivial symmetry group, each of which has $2p$ arrangements.

Note that $\frac{1}{2}(p-1)! - u - (u!2^{u-1} - u)p$ is divisible by $2p$ since

$$\tfrac{1}{2}(p-1)! - u(p-1) = \tfrac{1}{2}\left\{(p-1)! - (p-1)^2\right\}$$

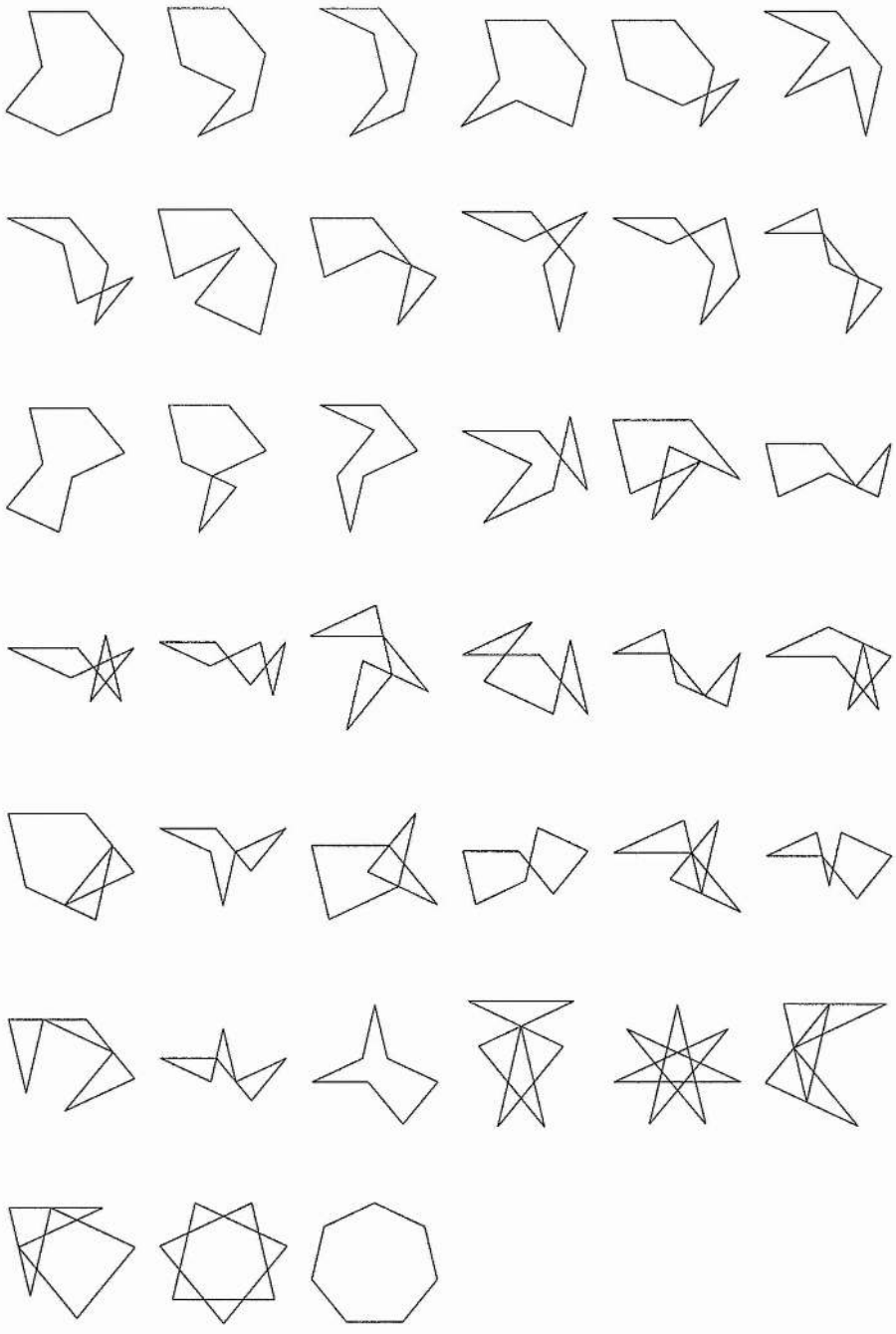
which is 0 modulo p by Wilson's Theorem.

Thus we can count the number of different closed shapes that can be generated by a PGP evaluated modulo a prime for given values of p (see Figure 9.2).

Example

See Figure 9.3 as an example of all the essentially different shapes available from a PGP with $p^m = 7$.

Figure 9.3 The 39 'essentially different' shapes possible from a PGP evaluated modulo 7.



Conclusion

Summary

We have presented methods that can predict the symmetry group of a PGP, $\mathcal{P}_{f,p^m}$ given $f \in \mathbb{Z}[x]$ and $p^m \in \mathbf{P}$. Because of the guaranteed periodic nature of the sequence generated by $f(j)$ evaluated modulo p^m for $j = 0, 1, 2, \ldots$ we can classify bounded PGPs as closed, and unbounded as open.

For bounded PGPs we have presented a complete set of procedures for pre-calculating the symmetry group. For unbounded PGPs we have gone some way to classifying the PGP into one of the seven frieze groups. However, because of the unsatisfactory nature of the reflection in a line parallel to the translation, we have not shown how this symmetry can always be predicted.

We have provided means to classify all polynomials into their coset of $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$, and in particular ways of testing whether a given polynomial when evaluated over $\mathbb{Z}$ is always congruent to 0 modulo p^m, even if f is not the zero polynomial in $\mathbb{Z}_{p^m}[x]$.

We have also seen how possible shapes are largely, though not exactly, equivalent to p^m-periodic sequences and that the question of generating particular shapes is similar to that of generating related periodic sequences modulo a prime power. In our investigations into constructing the shapes associated with the feasible sequences

we have seen how the iterated forward differences of p^m-periodic sequences evaluated modulo p^m disappear (*i.e.* converge to the zero sequence) sooner than might be anticipated, and how this allows for a host of identities for alternating sums of binomial coefficients evaluated modulo prime powers.

Comparing the iterated forward differences of p^m-periodic sequences with that of polynomials in $\mathbb{Z}[x]/\mathcal{Z}_{p^m,x}$, we have seen that certain sequences are unavailable through polynomial evaluation, and we have indicated how it is possible to raise the power on the prime in order to allow more flexibility in the available polynomials, resulting in a desired shape being generated by a different sequence than might initially have been thought the most appropriate.

Finally we have applied some of the theorems of Part II to the special case of f being a quadratic, and seen how striking images with cyclic symmetry groups (though mostly not dihedral) can be made through a PGP. The special case when the PGP is calculated evaluating polynomials modulo a prime is also investigated. This more straightforward case, where we are working over a field, $\mathbb{Z}_p$, means that all possible sequences are available through polynomials, and we look particularly at finding the number of essentially different closed shapes that can be generated modulo p.

In the appendices can be found many examples of PGPs with f the cubic ax^3, for various values of a with f evaluated modulo n for a range of values of n. A script is given that will generate encapsulated POSTSCRIPT files of the PGP specified in terms of f and n, with many options to vary the appearance of the PGP. We also present the procedures developed in Parts II and III as genuine working code for

a freely available mathematical package.

Final words

The ideas behind the methods of reconstruction given in Part III could be tightened somewhat to provide a less 'hit-and-miss' method of 'interpolating' a polynomial modulo p^m in order to generate a given shape.

Variants on the construction presented here have been investigated elsewhere (in particular [1, 8], and many of the theorems and procedures contained in this thesis could be adapted to suit these variations.

Appendices

Appendix A

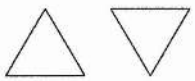
Cubic Examples

Figure A.1 A collection of PGPs with $n = 2, 3, \ldots, 29$ and with $f(x) = ax^3$ for $a = 1, 2, \ldots, n-1$.

$n = 2$

$\mathcal{P}_{1x^3,2}$

$n = 3$



$\mathcal{P}_{1x^3,3}$ $\mathcal{P}_{2x^3,3}$

$n = 4$

$\mathcal{P}_{1x^3,4}$ $\mathcal{P}_{2x^3,4}$ $\mathcal{P}_{3x^3,4}$

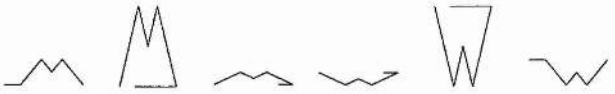
$n = 5$

$\mathcal{P}_{1x^3,5}$ $\mathcal{P}_{2x^3,5}$ $\mathcal{P}_{3x^3,5}$ $\mathcal{P}_{4x^3,5}$

$n = 6$

$\mathcal{P}_{1x^3,6}$ $\mathcal{P}_{2x^3,6}$ $\mathcal{P}_{3x^3,6}$ $\mathcal{P}_{4x^3,6}$ $\mathcal{P}_{5x^3,6}$

$n = 7$



$\mathcal{P}_{1x^3,7}$ $\mathcal{P}_{2x^3,7}$ $\mathcal{P}_{3x^3,7}$ $\mathcal{P}_{4x^3,7}$ $\mathcal{P}_{5x^3,7}$ $\mathcal{P}_{6x^3,7}$

$n = 8$

$\mathcal{P}_{1x^3,8}$ $\mathcal{P}_{2x^3,8}$ $\mathcal{P}_{3x^3,8}$ $\mathcal{P}_{4x^3,8}$ $\mathcal{P}_{5x^3,8}$ $\mathcal{P}_{6x^3,8}$ $\mathcal{P}_{7x^3,8}$

$n = 9$

$\mathcal{P}_{1x^3,9}$ $\mathcal{P}_{2x^3,9}$ $\mathcal{P}_{3x^3,9}$ $\mathcal{P}_{4x^3,9}$ $\mathcal{P}_{5x^3,9}$ $\mathcal{P}_{6x^3,9}$ $\mathcal{P}_{7x^3,9}$ $\mathcal{P}_{8x^3,9}$

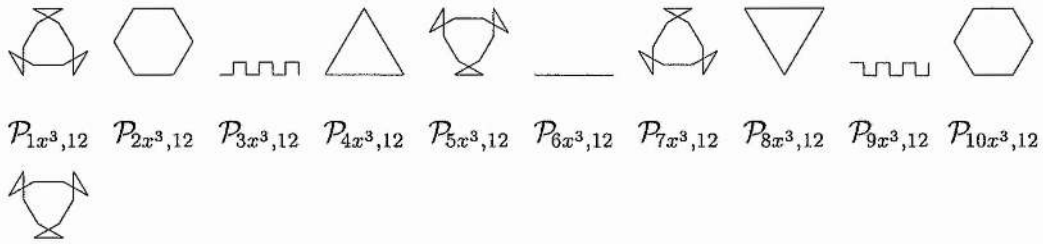
$n = 10$

$\mathcal{P}_{1x^3,10}$ $\mathcal{P}_{2x^3,10}$ $\mathcal{P}_{3x^3,10}$ $\mathcal{P}_{4x^3,10}$ $\mathcal{P}_{5x^3,10}$ $\mathcal{P}_{6x^3,10}$ $\mathcal{P}_{7x^3,10}$ $\mathcal{P}_{8x^3,10}$ $\mathcal{P}_{9x^3,10}$

$n = 11$

$\mathcal{P}_{1x^3,11}$ $\mathcal{P}_{2x^3,11}$ $\mathcal{P}_{3x^3,11}$ $\mathcal{P}_{4x^3,11}$ $\mathcal{P}_{5x^3,11}$ $\mathcal{P}_{6x^3,11}$ $\mathcal{P}_{7x^3,11}$ $\mathcal{P}_{8x^3,11}$ $\mathcal{P}_{9x^3,11}$ $\mathcal{P}_{10x^3,11}$

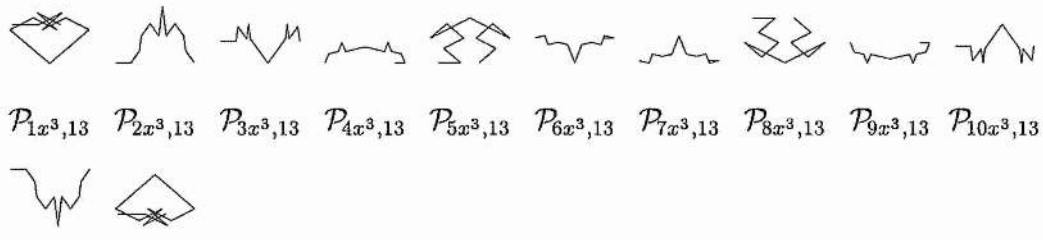
$n = 12$



$\mathcal{P}_{1x^3,12}$ $\mathcal{P}_{2x^3,12}$ $\mathcal{P}_{3x^3,12}$ $\mathcal{P}_{4x^3,12}$ $\mathcal{P}_{5x^3,12}$ $\mathcal{P}_{6x^3,12}$ $\mathcal{P}_{7x^3,12}$ $\mathcal{P}_{8x^3,12}$ $\mathcal{P}_{9x^3,12}$ $\mathcal{P}_{10x^3,12}$

$\mathcal{P}_{11x^3,12}$

$n = 13$



$\mathcal{P}_{1x^3,13}$ $\mathcal{P}_{2x^3,13}$ $\mathcal{P}_{3x^3,13}$ $\mathcal{P}_{4x^3,13}$ $\mathcal{P}_{5x^3,13}$ $\mathcal{P}_{6x^3,13}$ $\mathcal{P}_{7x^3,13}$ $\mathcal{P}_{8x^3,13}$ $\mathcal{P}_{9x^3,13}$ $\mathcal{P}_{10x^3,13}$

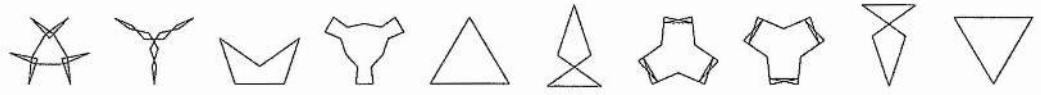
$\mathcal{P}_{11x^3,13}$ $\mathcal{P}_{12x^3,13}$

$n = 14$

$\mathcal{P}_{1x^3,14}$ $\mathcal{P}_{2x^3,14}$ $\mathcal{P}_{3x^3,14}$ $\mathcal{P}_{4x^3,14}$ $\mathcal{P}_{5x^3,14}$ $\mathcal{P}_{6x^3,14}$ $\mathcal{P}_{7x^3,14}$ $\mathcal{P}_{8x^3,14}$ $\mathcal{P}_{9x^3,14}$ $\mathcal{P}_{10x^3,14}$

$\mathcal{P}_{11x^3,14}$ $\mathcal{P}_{12x^3,14}$ $\mathcal{P}_{13x^3,14}$

$n = 15$

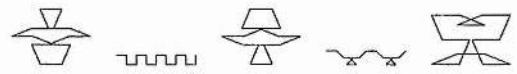


$\mathcal{P}_{1x^3,15}$ $\mathcal{P}_{2x^3,15}$ $\mathcal{P}_{3x^3,15}$ $\mathcal{P}_{4x^3,15}$ $\mathcal{P}_{5x^3,15}$ $\mathcal{P}_{6x^3,15}$ $\mathcal{P}_{7x^3,15}$ $\mathcal{P}_{8x^3,15}$ $\mathcal{P}_{9x^3,15}$ $\mathcal{P}_{10x^3,15}$

$\mathcal{P}_{11x^3,15}$ $\mathcal{P}_{12x^3,15}$ $\mathcal{P}_{13x^3,15}$ $\mathcal{P}_{14x^3,15}$

$n = 16$

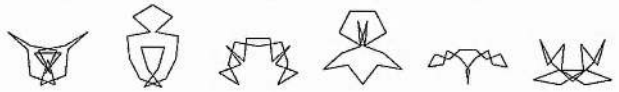
$\mathcal{P}_{1x^3,16}$ $\mathcal{P}_{2x^3,16}$ $\mathcal{P}_{3x^3,16}$ $\mathcal{P}_{4x^3,16}$ $\mathcal{P}_{5x^3,16}$ $\mathcal{P}_{6x^3,16}$ $\mathcal{P}_{7x^3,16}$ $\mathcal{P}_{8x^3,16}$ $\mathcal{P}_{9x^3,16}$ $\mathcal{P}_{10x^3,16}$



$\mathcal{P}_{11x^3,16}$ $\mathcal{P}_{12x^3,16}$ $\mathcal{P}_{13x^3,16}$ $\mathcal{P}_{14x^3,16}$ $\mathcal{P}_{15x^3,16}$

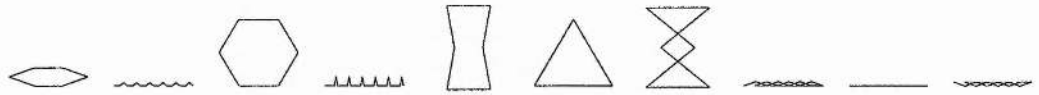
$n = 17$

$\mathcal{P}_{1x^3,17}$ $\mathcal{P}_{2x^3,17}$ $\mathcal{P}_{3x^3,17}$ $\mathcal{P}_{4x^3,17}$ $\mathcal{P}_{5x^3,17}$ $\mathcal{P}_{6x^3,17}$ $\mathcal{P}_{7x^3,17}$ $\mathcal{P}_{8x^3,17}$ $\mathcal{P}_{9x^3,17}$ $\mathcal{P}_{10x^3,17}$



$\mathcal{P}_{11x^3,17}$ $\mathcal{P}_{12x^3,17}$ $\mathcal{P}_{13x^3,17}$ $\mathcal{P}_{14x^3,17}$ $\mathcal{P}_{15x^3,17}$ $\mathcal{P}_{16x^3,17}$

$n = 18$

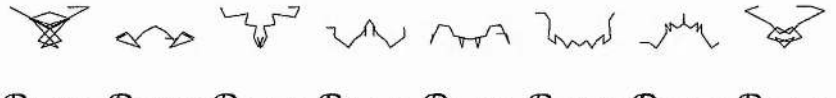


$\mathcal{P}_{1x^3,18}$ $\mathcal{P}_{2x^3,18}$ $\mathcal{P}_{3x^3,18}$ $\mathcal{P}_{4x^3,18}$ $\mathcal{P}_{5x^3,18}$ $\mathcal{P}_{6x^3,18}$ $\mathcal{P}_{7x^3,18}$ $\mathcal{P}_{8x^3,18}$ $\mathcal{P}_{9x^3,18}$ $\mathcal{P}_{10x^3,18}$

$\mathcal{P}_{11x^3,18}$ $\mathcal{P}_{12x^3,18}$ $\mathcal{P}_{13x^3,18}$ $\mathcal{P}_{14x^3,18}$ $\mathcal{P}_{15x^3,18}$ $\mathcal{P}_{16x^3,18}$ $\mathcal{P}_{17x^3,18}$

$n = 19$

$\mathcal{P}_{1x^3,19}$ $\mathcal{P}_{2x^3,19}$ $\mathcal{P}_{3x^3,19}$ $\mathcal{P}_{4x^3,19}$ $\mathcal{P}_{5x^3,19}$ $\mathcal{P}_{6x^3,19}$ $\mathcal{P}_{7x^3,19}$ $\mathcal{P}_{8x^3,19}$ $\mathcal{P}_{9x^3,19}$ $\mathcal{P}_{10x^3,19}$



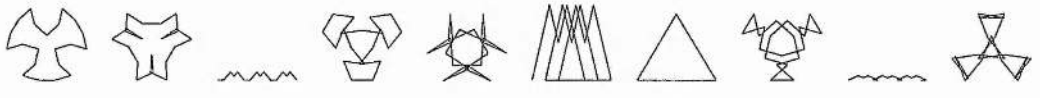
$\mathcal{P}_{11x^3,19}$ $\mathcal{P}_{12x^3,19}$ $\mathcal{P}_{13x^3,19}$ $\mathcal{P}_{14x^3,19}$ $\mathcal{P}_{15x^3,19}$ $\mathcal{P}_{16x^3,19}$ $\mathcal{P}_{17x^3,19}$ $\mathcal{P}_{18x^3,19}$

$n = 20$

$\mathcal{P}_{1x^3,20}$ $\mathcal{P}_{2x^3,20}$ $\mathcal{P}_{3x^3,20}$ $\mathcal{P}_{4x^3,20}$ $\mathcal{P}_{5x^3,20}$ $\mathcal{P}_{6x^3,20}$ $\mathcal{P}_{7x^3,20}$ $\mathcal{P}_{8x^3,20}$ $\mathcal{P}_{9x^3,20}$ $\mathcal{P}_{10x^3,20}$

$\mathcal{P}_{11x^3,20}$ $\mathcal{P}_{12x^3,20}$ $\mathcal{P}_{13x^3,20}$ $\mathcal{P}_{14x^3,20}$ $\mathcal{P}_{15x^3,20}$ $\mathcal{P}_{16x^3,20}$ $\mathcal{P}_{17x^3,20}$ $\mathcal{P}_{18x^3,20}$ $\mathcal{P}_{19x^3,20}$

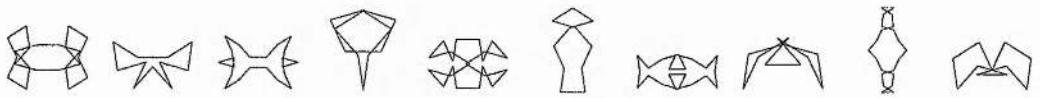
$n = 21$



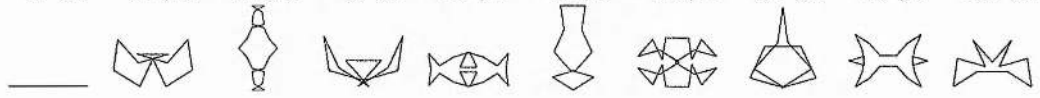
$\mathcal{P}_{1x^3,21}$ $\mathcal{P}_{2x^3,21}$ $\mathcal{P}_{3x^3,21}$ $\mathcal{P}_{4x^3,21}$ $\mathcal{P}_{5x^3,21}$ $\mathcal{P}_{6x^3,21}$ $\mathcal{P}_{7x^3,21}$ $\mathcal{P}_{8x^3,21}$ $\mathcal{P}_{9x^3,21}$ $\mathcal{P}_{10x^3,21}$

$\mathcal{P}_{11x^3,21}$ $\mathcal{P}_{12x^3,21}$ $\mathcal{P}_{13x^3,21}$ $\mathcal{P}_{14x^3,21}$ $\mathcal{P}_{15x^3,21}$ $\mathcal{P}_{16x^3,21}$ $\mathcal{P}_{17x^3,21}$ $\mathcal{P}_{18x^3,21}$ $\mathcal{P}_{19x^3,21}$ $\mathcal{P}_{20x^3,21}$

$n = 22$



$\mathcal{P}_{1x^3,22}$ $\mathcal{P}_{2x^3,22}$ $\mathcal{P}_{3x^3,22}$ $\mathcal{P}_{4x^3,22}$ $\mathcal{P}_{5x^3,22}$ $\mathcal{P}_{6x^3,22}$ $\mathcal{P}_{7x^3,22}$ $\mathcal{P}_{8x^3,22}$ $\mathcal{P}_{9x^3,22}$ $\mathcal{P}_{10x^3,22}$

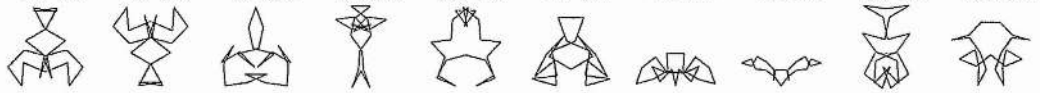


$\mathcal{P}_{11x^3,22}$ $\mathcal{P}_{12x^3,22}$ $\mathcal{P}_{13x^3,22}$ $\mathcal{P}_{14x^3,22}$ $\mathcal{P}_{15x^3,22}$ $\mathcal{P}_{16x^3,22}$ $\mathcal{P}_{17x^3,22}$ $\mathcal{P}_{18x^3,22}$ $\mathcal{P}_{19x^3,22}$ $\mathcal{P}_{20x^3,22}$

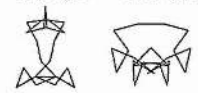
$\mathcal{P}_{21x^3,22}$

$n = 23$

$\mathcal{P}_{1x^3,23}$ $\mathcal{P}_{2x^3,23}$ $\mathcal{P}_{3x^3,23}$ $\mathcal{P}_{4x^3,23}$ $\mathcal{P}_{5x^3,23}$ $\mathcal{P}_{6x^3,23}$ $\mathcal{P}_{7x^3,23}$ $\mathcal{P}_{8x^3,23}$ $\mathcal{P}_{9x^3,23}$ $\mathcal{P}_{10x^3,23}$



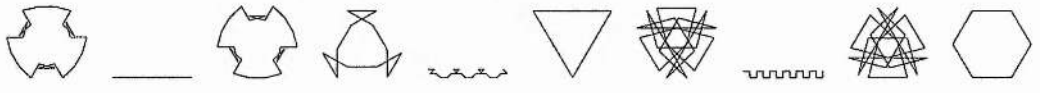
$\mathcal{P}_{11x^3,23}$ $\mathcal{P}_{12x^3,23}$ $\mathcal{P}_{13x^3,23}$ $\mathcal{P}_{14x^3,23}$ $\mathcal{P}_{15x^3,23}$ $\mathcal{P}_{16x^3,23}$ $\mathcal{P}_{17x^3,23}$ $\mathcal{P}_{18x^3,23}$ $\mathcal{P}_{19x^3,23}$ $\mathcal{P}_{20x^3,23}$



$\mathcal{P}_{21x^3,23}$ $\mathcal{P}_{22x^3,23}$

$n = 24$

$\mathcal{P}_{1x^3,24}$ $\mathcal{P}_{2x^3,24}$ $\mathcal{P}_{3x^3,24}$ $\mathcal{P}_{4x^3,24}$ $\mathcal{P}_{5x^3,24}$ $\mathcal{P}_{6x^3,24}$ $\mathcal{P}_{7x^3,24}$ $\mathcal{P}_{8x^3,24}$ $\mathcal{P}_{9x^3,24}$ $\mathcal{P}_{10x^3,24}$



$\mathcal{P}_{11x^3,24}$ $\mathcal{P}_{12x^3,24}$ $\mathcal{P}_{13x^3,24}$ $\mathcal{P}_{14x^3,24}$ $\mathcal{P}_{15x^3,24}$ $\mathcal{P}_{16x^3,24}$ $\mathcal{P}_{17x^3,24}$ $\mathcal{P}_{18x^3,24}$ $\mathcal{P}_{19x^3,24}$ $\mathcal{P}_{20x^3,24}$

$\mathcal{P}_{21x^3,24}$ $\mathcal{P}_{22x^3,24}$ $\mathcal{P}_{23x^3,24}$

$n = 25$

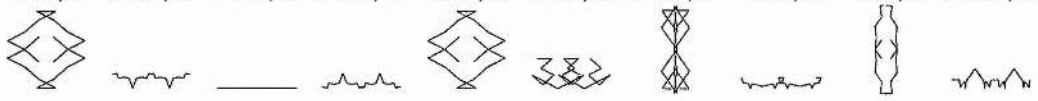
$\mathcal{P}_{1x^3,25}$ $\mathcal{P}_{2x^3,25}$ $\mathcal{P}_{3x^3,25}$ $\mathcal{P}_{4x^3,25}$ $\mathcal{P}_{5x^3,25}$ $\mathcal{P}_{6x^3,25}$ $\mathcal{P}_{7x^3,25}$ $\mathcal{P}_{8x^3,25}$ $\mathcal{P}_{9x^3,25}$ $\mathcal{P}_{10x^3,25}$

$\mathcal{P}_{11x^3,25}$ $\mathcal{P}_{12x^3,25}$ $\mathcal{P}_{13x^3,25}$ $\mathcal{P}_{14x^3,25}$ $\mathcal{P}_{15x^3,25}$ $\mathcal{P}_{16x^3,25}$ $\mathcal{P}_{17x^3,25}$ $\mathcal{P}_{18x^3,25}$ $\mathcal{P}_{19x^3,25}$ $\mathcal{P}_{20x^3,25}$

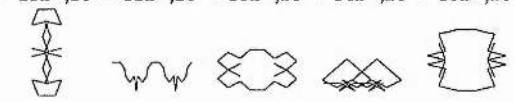
$\mathcal{P}_{21x^3,25}$ $\mathcal{P}_{22x^3,25}$ $\mathcal{P}_{23x^3,25}$ $\mathcal{P}_{24x^3,25}$

$n = 26$

$\mathcal{P}_{1x^3,26}$ $\mathcal{P}_{2x^3,26}$ $\mathcal{P}_{3x^3,26}$ $\mathcal{P}_{4x^3,26}$ $\mathcal{P}_{5x^3,26}$ $\mathcal{P}_{6x^3,26}$ $\mathcal{P}_{7x^3,26}$ $\mathcal{P}_{8x^3,26}$ $\mathcal{P}_{9x^3,26}$ $\mathcal{P}_{10x^3,26}$



$\mathcal{P}_{11x^3,26}$ $\mathcal{P}_{12x^3,26}$ $\mathcal{P}_{13x^3,26}$ $\mathcal{P}_{14x^3,26}$ $\mathcal{P}_{15x^3,26}$ $\mathcal{P}_{16x^3,26}$ $\mathcal{P}_{17x^3,26}$ $\mathcal{P}_{18x^3,26}$ $\mathcal{P}_{19x^3,26}$ $\mathcal{P}_{20x^3,26}$

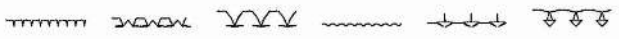


$\mathcal{P}_{21x^3,26}$ $\mathcal{P}_{22x^3,26}$ $\mathcal{P}_{23x^3,26}$ $\mathcal{P}_{24x^3,26}$ $\mathcal{P}_{25x^3,26}$

$n = 27$

$\mathcal{P}_{1x^3,27}$ $\mathcal{P}_{2x^3,27}$ $\mathcal{P}_{3x^3,27}$ $\mathcal{P}_{4x^3,27}$ $\mathcal{P}_{5x^3,27}$ $\mathcal{P}_{6x^3,27}$ $\mathcal{P}_{7x^3,27}$ $\mathcal{P}_{8x^3,27}$ $\mathcal{P}_{9x^3,27}$ $\mathcal{P}_{10x^3,27}$

$\mathcal{P}_{11x^3,27}$ $\mathcal{P}_{12x^3,27}$ $\mathcal{P}_{13x^3,27}$ $\mathcal{P}_{14x^3,27}$ $\mathcal{P}_{15x^3,27}$ $\mathcal{P}_{16x^3,27}$ $\mathcal{P}_{17x^3,27}$ $\mathcal{P}_{18x^3,27}$ $\mathcal{P}_{19x^3,27}$ $\mathcal{P}_{20x^3,27}$



$\mathcal{P}_{21x^3,27}$ $\mathcal{P}_{22x^3,27}$ $\mathcal{P}_{23x^3,27}$ $\mathcal{P}_{24x^3,27}$ $\mathcal{P}_{25x^3,27}$ $\mathcal{P}_{26x^3,27}$

$n = 28$

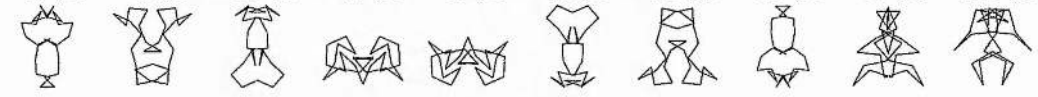
$\mathcal{P}_{1x^3,28}$ $\mathcal{P}_{2x^3,28}$ $\mathcal{P}_{3x^3,28}$ $\mathcal{P}_{4x^3,28}$ $\mathcal{P}_{5x^3,28}$ $\mathcal{P}_{6x^3,28}$ $\mathcal{P}_{7x^3,28}$ $\mathcal{P}_{8x^3,28}$ $\mathcal{P}_{9x^3,28}$ $\mathcal{P}_{10x^3,28}$

$\mathcal{P}_{11x^3,28}$ $\mathcal{P}_{12x^3,28}$ $\mathcal{P}_{13x^3,28}$ $\mathcal{P}_{14x^3,28}$ $\mathcal{P}_{15x^3,28}$ $\mathcal{P}_{16x^3,28}$ $\mathcal{P}_{17x^3,28}$ $\mathcal{P}_{18x^3,28}$ $\mathcal{P}_{19x^3,28}$ $\mathcal{P}_{20x^3,28}$

$\mathcal{P}_{21x^3,28}$ $\mathcal{P}_{22x^3,28}$ $\mathcal{P}_{23x^3,28}$ $\mathcal{P}_{24x^3,28}$ $\mathcal{P}_{25x^3,28}$ $\mathcal{P}_{26x^3,28}$ $\mathcal{P}_{27x^3,28}$

$n = 29$

$\mathcal{P}_{1x^3,29}$ $\mathcal{P}_{2x^3,29}$ $\mathcal{P}_{3x^3,29}$ $\mathcal{P}_{4x^3,29}$ $\mathcal{P}_{5x^3,29}$ $\mathcal{P}_{6x^3,29}$ $\mathcal{P}_{7x^3,29}$ $\mathcal{P}_{8x^3,29}$ $\mathcal{P}_{9x^3,29}$ $\mathcal{P}_{10x^3,29}$



$\mathcal{P}_{11x^3,29}$ $\mathcal{P}_{12x^3,29}$ $\mathcal{P}_{13x^3,29}$ $\mathcal{P}_{14x^3,29}$ $\mathcal{P}_{15x^3,29}$ $\mathcal{P}_{16x^3,29}$ $\mathcal{P}_{17x^3,29}$ $\mathcal{P}_{18x^3,29}$ $\mathcal{P}_{19x^3,29}$ $\mathcal{P}_{20x^3,29}$

$\mathcal{P}_{21x^3,29}$ $\mathcal{P}_{22x^3,29}$ $\mathcal{P}_{23x^3,29}$ $\mathcal{P}_{24x^3,29}$ $\mathcal{P}_{25x^3,29}$ $\mathcal{P}_{26x^3,29}$ $\mathcal{P}_{27x^3,29}$ $\mathcal{P}_{28x^3,29}$

Appendix B

PGP Perl Script

A Perl script that generates an Encapsulated PostScript file for a PGP. The script will accept a number of command line options to vary the 'amount' and appearance of the PGP generated. This script was used to generate the diagrams in this thesis.

Perl is a free scripting language interpreter available for most popular operating systems and system architectures.

See `http://www.perl.com/` for more details.

Lines ending with "\" signify a single line split for display purposes over multiple lines. The "\" (and the following line indentation) is not part of the actual line.

```
#!/usr/bin/perl

use Math::BigFloat;
use Getopt::Long;
$Getopt::Long::autoabbrev = 0;
$Getopt::Long::ignorecase = 0;

&Main();

sub Main {
    open(PGPLOG, ">>pgp.log");
    $date = qx(date);
    print PGPLOG "pgp @ARGV\n";
    close(PGPLOG);

    &GetandSetOptionsandVariables;
    &Sequences;
    &Boundaries;
    &WritePSFile;
}

sub SetDefaults {
```

```
# boolean defaults
    $AXES = 0;
    $BORDER = 0;
    $DASHED = 0;
    $VERTICES = 0;
    $EDGES = 1;
    $ELABELS = 0;
    $VLABELS = 0;
    $SPACE = 1;
    $ASK = 0;

# other defaults
    $width = 1.3;
    $maxheight = 8;
    $pgplinewidth = 0.3;
    $axeslinewidth = 0.5;
    $boxlinewidth = 2;
    $axesarrows = 1;
    $edgearrows = 0;
    $arrowheadwidth = 3;
    $arrowheadheight = 8;
    $edgearrowlength = 0.1;
    $edgearrowangle = 30;
    $vlabeldist = 0.3;
    $elabeldist = 0.3;
    $axesgray = 0.7;
    $boxgray = 0;
    $pgpgray = 0;
    $vertexgray = 0;
    $dotradius = 2;
    $defaultaxesextra = 10;
    $defaultborderspace = 8;
    $dash1 = 0.2;
    $dash2 = 0;
    $fontsize = 8;
    $linecap = 1;
    $variable = "x";
    $numdots = 20;
    $report = 1;
}

sub Sequences {
# calculate i sequences
    $dotcount = 0;
    if ($report) { print STDERR "Calculating sequence "; }
    for $i ( (0..$end) ) {
        if ($report) { &CheckDot; }
        $imodn = &Math::BigInt::bmod($i,$n);
        ($fv = $fx) =~ s/x/($imodn)/g;
        $fvmodn = (&Math::BigInt::bmod(eval($fv),$n));
        push(@seq, $fvmodn);
    }
    if ($nend) {
        if ($report) { &CheckDot; }
        @nseq = ();
        @nxseq = (0,0);
        @nyseq = (0,0);
        for $i ( ($nbegin..$nend) ) {
            $imodn = &Math::BigInt::bmod($i,$n);
            ($fv = $fx) =~ s/x/(-$imodn-1)/g;
            $fvmodn = (&Math::BigInt::bmod(eval($fv),$n));
            push(@nseq, $fvmodn);
        }
    }
    if ($report) { print STDERR "\nCalculating positions"; }
# calculate x and y sequences
    $dotcount = 0;
    $c = $begin;
    $x = $y = 0;
    for $f (@seq) {
        if ($report) { &CheckDot; }
        $fi = (2*$PI/$n)*$f;
```

```
        if ($c == 0) { $Mx = $mx = $x; $My = $my = $y; }
        $c --;
        $x += cos($fi);
        $y += sin($fi);
        if ($x >= $Mx) { $Mx = $x; }
        if ($y >= $My) { $My = $y; }
        if ($x <= $mx) { $mx = $x; }
        if ($y <= $my) { $my = $y; }
        push(@xseq, $x);
        push(@yseq, $y);
    }
    if ($nend) {
        $c = 0;
        $x = $y = 0;
        for $f (@nseq) {
            if ($report) { &CheckDot; }
            $fi = (2*$PI/$n)*$f;
            $c ++;
            $x -= cos($fi);
            $y -= sin($fi);
            if ($c < $nend) {
                if ($x >= $Mx) { $Mx = $x; }
                if ($y >= $My) { $My = $y; }
                if ($x <= $mx) { $mx = $x; }
                if ($y <= $my) { $my = $y; }
            }
            push(@nxseq, $x);
            push(@nyseq, $y);
        }
    }
}

sub Boundaries {
# calculate boundaries
    $pgpwidth = $width - 2*($borderspace+$axesextra);
    $maxpgpheight = $maxheight - 2*($borderspace+$axesextra);
    if (($My-$my)*$pgpwidth/($Mx-$mx) > $maxpgpheight) {
        $scale = $maxpgpheight/($My-$my);
    } else {
        $scale = $pgpwidth/($Mx-$mx);
    }
    $Mx *= $scale;
    $My *= $scale;
    $mx *= $scale;
    $my *= $scale;
    if (($Mx-$mx) < $pgpwidth) {
        $oldMx = $Mx;
        $Mx = int($Mx*$pgpwidth/($Mx-$mx));
        $mx = int($mx*$pgpwidth/($oldMx-$mx));
    }
    $elabeldist *= $scale;
    $vlabeldist *= $scale;
    $edgearrowlength *= $scale;
    $dash1 *= $scale;
    $dash2 *= $scale;
    $aMx = int($Mx + $axesextra);
    $aMy = int($My + $axesextra);
    $amx = int($mx - $axesextra);
    $amy = int($my - $axesextra);
    $bMx = int($aMx + $borderspace);
    $bMy = int($aMy + $borderspace);
    $bmx = int($amx - $borderspace);
    $bmy = int($amy - $borderspace);
}

sub WritePSFile {
    open(PSFILE, ">$filename");
    print PSFILE <<EOH;
%!PS-Adobe-3.0 EPSF-3.0
%%Title:       $title
%%Creator:     pgp
%%Pages:       1
```

```
%%BoundingBox: $bmx $bmy $bMx $bMy
%%DocumentNeededResources: font Helvetica
%%EndComments

/stringbbox {gsave newpath 0 0 moveto false charpath flattenpath
   pathbbox 4 2 roll pop pop 1.1 mul cvi exch 1.1 mul
   cvi exch grestore} def

/Helvetica findfont $fontsize scalefont setfont
% Usage for dirlabel: (string) distance angle x y dirlabel
/dirlabel {gsave translate dup rotate exch 0 translate -1 mul    \
    rotate dup stringbbox -2 div exch -2 div exch moveto show    \
    grestore} def
/edgearrow {gsave translate rotate $edgearrowlength 180          \
    $edgearrowangle sub sin mul 2 div 0 translate newpath 0 0    \
    moveto gsave 180 $edgearrowangle sub rotate $edgearrowlength\
     0 lineto stroke grestore newpath 0 0 moveto 180             \
    $edgearrowangle add rotate $edgearrowlength 0 lineto stroke \
    grestore} def
/right 0 def
/upright 45 def
/up 90 def
/upleft 135 def
/left 180 def
/downleft 225 def
/down 270 def
/downright 315 def
/vdist $elabeldist def
/edist $vlabeldist def

$linecap setlinecap

EOH
    ;

    if ($AXES) {
        $xb = $aMy - $arrowheadheight;
        $yb = $aMx - $arrowheadheight;
        print PSFILE "
%AXES
$axesgray setgray  $axeslinewidth setlinewidth
newpath  0 $amy moveto  0 $aMy lineto  stroke
newpath  $amx 0 moveto  $aMx 0 lineto  stroke
";
        if ($axesarrows) {
            print PSFILE "
newpath  -$arrowheadwidth $xb moveto 0 $aMy lineto               \
    $arrowheadwidth $xb lineto fill
newpath  $yb $arrowheadwidth moveto $aMx 0 lineto $yb            \
    -$arrowheadwidth lineto fill
";
        }
    }

    if ($report) { print STDERR "\nWriting PS file      "; }
    print PSFILE "%PGP\n$pgpgray setgray  $pgplinewidth           \
    setlinewidth\nnewpath\n";
#0 AND POSITIVES
    $c = 0;
    push(@seq,"b");
    while(@seq) {
        if ($report) { &CheckDot; }
        $i = shift(@seq);
        $sx = shift(@xseq)*$scale;
        $sy = shift(@yseq)*$scale;
        if ($c == $begin) {
            $lastsx = $sx;
            $lastsy = $sy;
            if ($VERTICES) {
                print PSFILE "newpath $sx $sy $dotradius 0 360  \
    arc fill\n";
            }
```

```
        if ($VLABELS) {
            $mul = "";
            if ($ASK) {
                print STDERR "Direction for V$c label?      \
[\"-\",\"\",\"n\"]: ";
                $mul = <STDIN>;
                chomp $mul;
            }
            $vdir = "down";
            if ($mul =~ /^$|^-$/) {
                print PSFILE "(V$c) ${mul}1 vdist mul $vdir \
$sx $sy dirlabel\n";
            }
            if ($mul =~ /N/) {
                $VLABELS = 0;
            }
        }
    }
    if ($c > $begin) {
        if (!defined($xseq[0]) && $DASHED) {
            print PSFILE "[$dash1] $dash2 setdash\n";
        }
        if ($EDGES) {
            print PSFILE "newpath $lastsx $lastsy moveto $sx\
 $sy lineto stroke\n";
        }
        ($lsx,$lsy) = (($lastsx + $sx)/2,($lastsy + $sy)/2);
        if ($edgearrows && (($DASHED && $c-1 < $end) ||      \
!$DASHED)) {
            $adir = $lasti*360/$n;
            print PSFILE "$adir $lsx $lsy edgearrow\n";
        }
        if ($ELABELS && (($DASHED && $c-1 < $end) ||         \
!$DASHED)) {
            $mul = "";
            $lastc = $c-1;
            if ($ASK) {
                print STDERR "Direction for E$lastc label?  \
[\"-\",\"\",\"n\"]: ";
                $mul = <STDIN>;
                chomp $mul;
            }
            $avg = ($lasti + $n/4);
            $edir = $avg*360/$n;
            if ($mul =~ /^$|^-$/) {
                print PSFILE "(E$lastc) ${mul}1 edist mul    \
$edir $lsx $lsy dirlabel\n";
            }
            if ($mul =~ /N/) {
                $ELABELS = 0;
            }
        }
        if ($VERTICES && defined($xseq[0])) {
            print PSFILE "newpath $sx $sy $dotradius 0 360  \
arc fill\n";
        }
        if ($VLABELS && defined($xseq[0])) {
            $mul = "";
            if ($ASK) {
                print STDERR "Direction for V$c label?      \
[\"-\",\"\",\"n\"]: ";
                $mul = <STDIN>;
                chomp $mul;
            }
            $avg = ($lasti + $i + $n/2)/2;
            $vdir = $avg*360/$n;
            if ($mul =~ /^$|^-$/) {
                print PSFILE "(V$c) ${mul}1 vdist mul $vdir \
$sx $sy dirlabel\n";
            }
            if ($mul =~ /N/) {
                $VLABELS = 0;
```

```
                }
            }
        }
        $lastsx = $sx;
        $lastsy = $sy;
        $lasti = $i;
        $c ++;
    }
#NEGATIVES
    if ($nend) {
        $c = 0;
        push(@nseq,"b");
        while(@nseq) {
            if ($report) { &CheckDot; }
            $i = shift(@nseq);
            $sx = shift(@nxseq)*$scale;
            $sy = shift(@nyseq)*$scale;
            if ($c == $nbegin) {
                $nextsx = $sx;
                $nextsy = $sy;
                if ($VERTICES) {
                    print PSFILE "[] 0 setdash\nnewpath $sx $sy \
    $dotradius 0 360 arc fill\n";
                }
            }
            if ($c > $nbegin) {
                if (!defined($nxseq[1]) && $DASHED) {
                    print PSFILE "[$dash1] $dash2 setdash\n";
                }
                if ($EDGES) {
                    print PSFILE "newpath $nextsx $nextsy moveto\
     $sx $sy lineto stroke\n";
                }
                if ($ELABELS && (($DASHED && $c-1 < $nend) ||   \
    !$DASHED) && $c>1) {
                    $mul = "";
                    $nextc = $c-1;
                    if ($ASK) {
                        print STDERR "Direction for E-$nextc    \
    label? [\"-\",\"\",\"n\"]: ";
                        $mul = <STDIN>;
                        chomp $mul;
                    }
                    ($lsx,$lsy) = (($nextsx + $sx)/2,($nextsy + \
    $sy)/2);
                    $avg = ($nextagaini + $n/4);
                    $edir = $avg*360/$n;
                    if ($mul =~ /^$|^-$/) {
                        print PSFILE "(E-$nextc) ${mul}1 edist  \
    mul $edir $lsx $lsy dirlabel\n";
                    }
                    if ($mul =~ /N/) {
                        $ELABELS = 0;
                    }
                }
                if ($VERTICES && defined($nxseq[1])) {
                    print PSFILE "newpath $sx $sy $dotradius 0  \
    360 arc fill\n";
                }
                if ($VLABELS && defined($nxseq[1])) {
                    $mul = "";
                    if ($ASK) {
                        print STDERR "Direction for V-$nextc    \
    label? [\"-\",\"\",\"n\"]: ";
                        $mul = <STDIN>;
                        chomp $mul;
                    }
                    $avg = ($nextagaini + $nexti + $n/2)/2;
                    $vdir = $avg*360/$n;
                    $nextc = $c-1;
                    if ($mul =~ /^$|^-$/ && $nextc > 0) {
                        print PSFILE "(V-$nextc) ${mul}1 vdist  \
```

```
    mul $vdir $sx $sy dirlabel\n";
                    }
                    if ($mul =~ /N/) {
                        $VLABELS = 0;
                    }
                }
            }
            $nextagainsx = $nextsx;
            $nextagainsy = $nextsy;
            $nextsx = $sx;
            $nextsy = $sy;
            $nextagaini = $nexti;
            $nexti = $i;
            $c ++;
        }
    }
    if ($BORDER) {
        print PSFILE "%BORDER\n[] 0 setdash\n$boxgray setgray   \
    $boxlinewidth setlinewidth\nnewpath\n$bmx $bmy moveto\n$bMx \
    $bmy lineto\n$bMx $bMy lineto\n$bmx $bMy lineto\n$bmx $bmy  \
    lineto\nstroke\n\n";
    }

    print PSFILE <<EOF;
showpage
%%EOF
EOF
    ;

    close(PSFILE);
    if ($report) { print STDERR "\nSuccessfully wrote PostScript\
     file $filename !\n"; }
}

sub GetandSetOptionsandVariables {
# constants
    $PI = 3.14159265358979323846;

# pre option defaults
    &SetDefaults;

# get word options
    &GetOptions(
                "polynomial=s" => \$fx,
                "p=s" => \$fx,
                "mod=i" => \$n,
                "m=i" => \$n,
                "repeats=i" => \$repeats,
                "range=s" => \$range,
                "r=s" => \$range,
                "begin=i" => \$begin,
                "end=i" => \$end,
                # get boolean options
                "all" => \$ALL,
                "A" => \$ALL,
                "axes!" => \$AXES,
                "a!" => \$AXES,
                "border!" => \$BORDER,
                "b!" => \$BORDER,
                "dashedends!" => \$DASHED,
                "d!" => \$DASHED,
                "vertices!" => \$VERTICES,
                "v!" => \$VERTICES,
                "edges!" => \$EDGES,
                "e!" => \$EDGES,
                "vlabels!" => \$VLABELS,
                "vl!" => \$VLABELS,
                "elabels!" => \$ELABELS,
                "el!" => \$ELABELS,
                "space!" => \$SPACE,
                "s!" => \$SPACE,
                "ask!" => \$ASK,
```

```
                # get other options
                "width=f" => \$width,
                "w=f" => \$width,
                "maxheight=f" => \$maxheight,
                "mh=f" => \$maxheight,
                "title=s" => \$title,
                "t=s" => \$title,
                "filename=s" => \$filename,
                "f=s" => \$filename,
                "edgewidth=f" => \$pgplinewidth,
                "ew=f" => \$pgplinewidth,
                "borderspace=f" => \$borderspace,
                "bs=f" => \$borderspace,
                "axesspace=f" => \$axesextra,
                "as=f" => \$axesextra,
                "arrowwidth=f" => \$arrowheadwidth,
                "Aw=f" => \$arrowheadwidth,
                "arrowheight=f" => \$arrowheadheight,
                "Ah=f" => \$arrowheadheight,
                "axesgray=f" => \$axesgray,
                "ag=f" => \$axesgray,
                "bordergray=f" => \$boxgray,
                "bg=f" => \$boxgray,
                "edgegray=f" => \$pgpgray,
                "eg=f" => \$pgpgray,
                "vertexgray=f" => \$vertexgray,
                "vg=f" => \$vertexgray,
                "axeswidth=f" => \$axeslinewidth,
                "aw=f" => \$axeslinewidth,
                "borderwidth=f" => \$boxlinewidth,
                "bw=f" => \$boxlinewidth,
                "vertexradius=f" => \$dotradius,
                "vr=f" => \$dotradius,
                "dash1=f" => \$dash1,
                "d1=f" => \$dash1,
                "dash2=f" => \$dash2,
                "d2=f" => \$dash2,
                "vlabeldist=f" => \$vlabeldist,
                "vld=f" => \$vlabeldist,
                "elabeldist=f" => \$elabeldist,
                "eld=f" => \$elabeldist,
                "fontsize=f" => \$fontsize,
                "fs=f" => \$fontsize,
                "linecap=f" => \$linecap,
                "lc=f" => \$linecap,
                "numdots=s" => \$numdots,
                "nd=s" => \$numdots,
                "report!" => \$report,
                "axesarrows!" => \$axesarrows,
                "aa!" => \$axesarrows,
                "edgearrows!" => \$edgearrows,
                "ea!" => \$edgearrows,
                "edgearrowlength=f" => \$edgearrowlength,
                "eal=f" => \$edgearrowlength,
                "edgearrowangle=f" => \$edgearrowangle,
                "eaa=f" => \$edgearrowangle,
                "variable=s" => \$variable,
                "x=s" => \$variable);

# post option defaults
# inches to points
    $width *= 72;
    $maxheight *= 72;
# timesavers
    if ($ALL) {
        $AXES = 1 unless $AXES == 0;
        $BORDER = 1 unless $BORDER == 0;
        $DASHED = 1 unless $DASHED == 0;
        $VERTICES = 1 unless $VERTICES == 0;
        $EDGES = 1 unless $EDGES == 0;
        $VLABELS = 1 unless $VLABELS == 0;
        $ELABELS = 1 unless $ELABELS == 0;
```

```
    }
# rangesetting
    if ( !(defined($begin) ** defined($end)) && $range =~       \
    /^(-?[0-9]*)((\.\.)|-)(-?[0-9]*)/) { $begin = $1; $end = $4;\
     }
    if ($range =~ /^[0-9]+$/) { $repeats = $range; }
    if ( !(defined($begin) && defined($end)) && $repeats ) {
        $begin = 0;
        $end = $repeats*$n-1;
    }
    $begin = 0 unless defined($begin);
    $end = $n-1 unless defined($end);
    $range = "$begin-$end";
# polynomial, title and filename
    $fx =~ s/[\*\^\s\"\']//g;
    $fxmodn = "${fx}_mod_$n:$range";
    $fx =~ s/$variable(\d)/x\*\*$1/g;
    $fx =~ s/(\d)$variable/$1\*x/g;
    $title = $fxmodn unless $title;
    $filename = "$fxmodn.eps" unless $filename;
# spacing
    if ($SPACE) {
        $borderspace = $defaultborderspace unless $borderspace;
        if ($AXES) {
            $axesextra = $defaultaxesextra unless $axesextra;
        }
    } else {
        $borderspace = 2*$pgplinewidth;
        if ($BORDER) {
            $borderspace += $boxlinewidth;
        }
        $axesextra = 0;
    }

# test enough valid options
    if (!($fx=~/^[0-9x\*\+\-\.eE]+$/ && $n=~/^[0-9]+$/ > 0 &&   \
    $n>0 && $begin=~/^-?[0-9]+$/ && $end=~/^[0-9]+$/)) {
        &PrintUsage;
        exit 0;
    }

# initialise other variables
    if ($begin<0) {
        $nbegin = 0;
        $nend = -$begin;
        $begin = 0;
    }

    @seq = ();
    @xseq = (0);
    @yseq = (0);
    @vlabelseq = (0);
    @elabelseq = ();

    if ($nend) {
        $dotspace = int(($end+$nend)/$numdots);
    } else {
        $dotspace = int($end/$numdots);
    }

}

sub OtherOptions {
    my $options = shift();
    $otheroptions .= "$options ";
}

sub PrintUsage {
    print STDERR<<EOUSAGE
Usage: pgp -p f(x) -m n [-r range] [options]
       -p, --polynomial
           f(x) a polynomial in 'x'.
```

```
        "*", "^", "**", " " are optional.
        e.g. 3x2+4x+1, '3*x^2 + 4*x + 1', 3*x**2+4*x+1
        are equivalent.
    -m, --mod
        n       a natural number.
    -r, --range [ | ( --begin b & --end e ) | --repeats r ]
        range   a range of values of the form b-e or b..e
        b can be negative.
        Alternatively a single number r can be used
        representing the range 0..m*n-1
        If a range is omitted 0..n-1 is assumed.
    boolean options
    to switch a boolean option off use --no<option>
    --axes,-a       draws on the x,y-axes.
                    default = off
    --border,-b     draws a bounding box.
                    default = off
    --dashed,-d     or DOTTED draws the end edges as dashed
                    lines (not E0) and misses out the end
                    vertices (not E0).
                    default = off
    --vertices,-v draws small circles at the vertices.
                    default = off
    --edges,-e      draws the edges.
                    default = on
    --vlabels,-V    labels the edges as V0 et c.
                    default = off
    --elabels,-E    labels the edges as E0 et c.
                    default = off
    --ask           interactively asks whether labels should
                    be on one side or the other.
                    Best to try without ASK first,
                    then run with --ask and press return
                    for labels on the correct side and
                    enter '-' for labels on the wrong
                    side.
                    Enter 'n' to omit a label.
                    Enter 'N' to omit all subsequent labels.
                    default = off
    --space,-s      leaves whitespace for axes (if on) and
                    border.
                    default = on
    --all,-A        all of the above except for
    --ask and --nospace.
    other options with arguments
    width in inches. Other distances are in points.
    --width
    -t, --title
    -f, --filename
    --edgewidth
    --borderspace
    --axesextra
    --arrowwidth
    --arrowheight
    --axesgray
    --bordergray
    --edgegray
    --vertexgray
    --axeswidth
    --borderwidth
    --vertexradius
    --dash1
    --dash2
    --vlabeldist
    --elabeldist
    --fontsize
    --linecap
    --variable
EOUSAGE
    ;
}
```

```
sub CheckDot {
    if ($dotcount == $dotspace) {
        print STDERR ".";
        $dotcount = 0;
    }
    $dotcount++;
}
```

Appendix C

GP/Pari Procedures

The procedures developed in pseudo-code in this thesis are presented here for the GP/Pari number theoretical computaion package. GP/Pari is available free as source or pre-compiled binaries for many popular operating systems and computer architectures.

See `http://hasse.mathematik.tu-muenchen.de/ntsw/pari/` for more details.

Procedures from the text

```
firstrep(f,p,m)=
{
 local(mm);
 for(mm=0,m,
  if(inZ(subst(f,x,x+p^mm)-f,p,m),return([p,mm]));
 )
}

isclosed(f,p,m)=
{
 local(mm,r,F,j);
 mm=firstrep(f,p,m)[2];
 r=p^mm;
 F=sum(j=0,r-1,x^(subst(f,x,j)%p^m));
 if(divrem(F,polcyclo(p^m))[2]==0,return(1),return(0))
}

rotsymm(f,p,m)=
{
 local(mm,i,c);
 mm=firstrep(f,p,m)[2];
 for(i=0,mm-1,
  c=(subst(f,x,p^i)-subst(f,x,0))%p^m;
  if(inZ(subst(f,x,x+p^i)-f-c,p,m),return([p,mm-i]));
 );
 return([p,0]);
}

hasrefsymm(f,p,m)=
{
 local(mm,rho,r,k);
 mm=firstrep(f,p,m)[2];
```

```
 rho=p^rotsymm(f,p,m)[2];
 r=p^mm/rho;
 for(k=0,r-1,
  if(inZ(subst(fdiff(f),x,k+x)-subst(fdiff(f),x,k-2-x),p,m)
       ||
      inZ(subst(fdiff(f),x,k-1+x)-subst(fdiff(f),x,k-x),p,m),
   return(1));
 );
 return(0);
}

minfactorial(p,m)=
{
 local(r,c,d);
 if(m==0,0,
  r=floor(logdiv(p,m));
  c=floor(m*(p-1)/(p^r-1));
  d=m-c*(p^r-1)/(p-1);
  c*p^r+minfactorial(p,d);
 )
}

reduce(f,p,m=1)=
{
 local(fb,i,j,s,k);
 fb=f;
 i=poldegree(f);
 for(j=0,m,
  s=minfactorial(p,m-j);
  for(k=0,i-s,
   fb=fb-p^j*floor(polcoeff(fb,i-k)/p^j)*x^(i-k-s)*zeropoly(s);
  );
  i=s;
 );
 fb;
}

inZ(f,p,m=1)=
{
 local(fb);
 fb=reduce(f,p,m);
 if(fb==0,return(1),return(0));
}

openhasrefsymm(f,p,m)=
{
 local(r,k);
 r=p^firstrep(f,p,m)[2];
 for(k=0,r-1,
  if(inZ(subst(fdiff(f),x,k+x)-subst(fdiff(f),x,k-2-x),p,m)
     || inZ(subst(fdiff(f),x,k-1+x)-subst(fdiff(f),x,k-x),p,m),print(k);
   return(1));
 );
 return(0);
}

openhasrotsymm(f,p,m)=
{
 local(r,k);
 r=p^firstrep(f,p,m)[2];
 for(k=0,r-1,
  if(inZ(subst(f,x,k+x)-subst(f,x,k-x-1),p,m)
     || inZ(subst(f,x,k+x)-subst(f,x,k-x),p,m),print(k);
   return(1));
 );
 return(0);
}

openhasglideref(f,p,m)=
{
 local(mm);
 if(p==2,
```

```
  mm=firstrep(f,p,m)[2];
  if(mm>0,return(inZ(subst(fdiff(f),x,x+p^(mm-1))+fdiff(f),p,m)));
 );
 return(0);
}

interp(A,p,m)=
{
 local(s,j,f,z);
 A=modseq(A,p,m);
 s=minfactorial(p,m);
 if(Dmn(A,p^m,s)==vector(p^m,j,0),
  f=0;
  for(j=0,s-1,
   z=zeropoly(j);
   if(ppower((subst(z,x,j)%p^m),p)<=ppower(((A[j+1]-subst(f,x,j))%p^m),p),
    f=f+z*(((A[j+1]-subst(f,x,j))/(subst(z,x,j)))%p^m);
   )
  );
  reduce(f,p,m);
 ,
  error("Doesn't iterate to [0,...] quicky enough");
 )
}
```

Helper procedures

```
fdiff(f)=
{
 return(subst(f,x,x+1)-f);
}

logdiv(p,m)=
{
 local(v,ld,r,pr,mr);
 v=factorint(m*p-m+1);
 ld=0;
 for(r=1,matsize(v)[1],
  pr=v[r,1];
  mr=v[r,2];
  if(pr==p,ld=ld+mr,ld=ld+mr*log(pr)/log(p));
 );
 ld
}

factorialppower(x,p)=
{
 local(i);
 sum(i=1,ceil(log(x)/log(p)),floor(x/p^i));
}

ppower(x,p)=
{
 local(fv);
 if(isprime(p),
  fv=factor(x);
  if(x%p==0&&x!=0,fv[setsearch(Set(vector(matsize(fv)[1],i,fv[i,1])),p),2],0);
 )
}

psum(p,r)=
{
 sum(i=0,r,p^i);
}

checkpp(pmax,mmax)=
{
 local(i,p,m,s,k);
 for(i=1,pmax,
```

```
  p=prime(i);
  print("p=",p);
  for(m=1,mmax,
   print1(".");
   s=maxdegree(p,m);
   if( (s!%p^m)==0 && ((s-1)!%p^m)!=0,
    ,
    print("");
    print("p=",p,", m=",m," NOT OK!!");
    for(k=1,m,
     print1(" ");
    )
   )
  );
  print("")
 )
}

zeropoly(s)=
{
 prod(i=0,s-1,x-i);
}

diff(f,g,p,m)=
{
 local(diffnum,j);
 diffnum=0;
 for(j=0,p^m-1,
  if(subst(f,x,j)%p^m != subst(g,x,j)%p^m,diffnum++)
 );
 diffnum;
}

displaypowers(f,p,m)=
{
 local(j,fx);
 for(j=0,p^m-1,
  fx=subst(f,x,j);
  if(fx==0,fx=p^m,);
  print(j,":\t",ppower(fx,p));
 )
}

U(n,i)=
{
 local(j);
 vector(n,j,
  if(j==i+1,
   1,
   0;
  )
 )
}

D(A)=
{
 local(j);
 vector(matsize(A)[2],j,
  if(j<matsize(A)[2],
   A[j+1]-A[j],
   A[1]-A[j];
  )
 )
}

Dn(A,n)=
{
 local(B,i);
 B=A;
 for(i=1,n,
  B=D(B)
 );
```

```
 B;
}

binom(k,l)=
 k!/(k-l)!/l!;

Dnentry(n,j,k)=
{
 local(l);
 sum(l=min(1,j),floor((k+j)/n),(-1)^(k-(n*l-j))*binom(k,n*l-j));
}

Dmnentry(n,j,k)=
{
 local(l);
 sum(l=min(1,j),floor((k+j)/n),(-1)^(k-(n*l-j))*binom(k,n*l-j))%n;
}

Dnrow(n,k)=
{
 local(j);
 vector(n,j,Dnentry(n,j-1,k));
}

Dmnrow(n,k)=
{
 local(j);
 vector(n,j,Dmnentry(n,j-1,k));
}

Dm(A,m)=
{
 local(j);
 vector(matsize(A)[2],j,
  if(j<matsize(A)[2],
   (A[j+1]-A[j])%m,
   (A[1]-A[j])%m;
  )
 )
}

Dmn(A,m,n)=
{
 local(B,i);
 B=A;
 for(i=1,n,
  B=Dm(B,m)
 );
 B;
}

displayDmnmatrix(p,m)=
{
 local(k,j);
 matrix(zeronum(p,m,m),p^m,k,j,Dmnentry(p^m,j-1,k-1));
}

printvec(A,n)=
{
 local(i,j);
 print1("[");
 for(i=1,matsize(A)[2],
  for(j=1,floor(log(n)/log(10))-floor(log(if(A[i]>0,A[i],1))/log(10)),
   print1(" ")
  );
  print1(A[i]);
  if(i<matsize(A)[2],
   print1(",")
  )
 );
 print("]");
}
```

```
display(p,m)=
{
 local(i);
 print(zeronum(p,m));
 for(i=0,zeronum(p,m),
  print1(i,":\t");
  printvec(Dmn(U(p^m,0),p^m,i),p^m);
 )
}

displaym(p,m,r)=
{
 local(i);
 print(zeronum(p,m,r));
 for(i=0,zeronum(p,m,r),
  print1(i,":\t");
  printvec(Dmn(U(p^m,0),p^r,i),p^r);
 )
}

zeronum(p,m,r)=
 p^m+(r-1)*(p-1)*p^(m-1);

vals(f,p,m)=
{
 local(j);
 print1("[");
 for(j=0,p^m-1,
  print1(subst(f,x,j)%p^m);
  if(j<p^m-1,print1(","))
 );
 print("]");
}

positivemod(n,m)=
 if(n%m==0,m,n%m);

modseq(A,p,m)=
{
 local(j);
 vector(p^m,j,A[positivemod(j,matsize(A)[2])]%p^m);
}

fvals(f,p,m)=
{
 local(j);
 vector(p^m,j,subst(f,x,j-1)%p^m);
}
```

References

[1] Harold Abelson and Andrea A. diSessa. *Turtle geometry : the computer as a medium for exploring mathematics.* MIT Press, Cambridge, Mass.; London, 1981.

[2] D. F. Bailey. More binomial coefficient congruences. *Fibonacci Quart.*, 30(2):121–125, 1992.

[3] H. S. M. Coxeter. *Introduction to Geometry.* Wiley, New York, 1961.

[4] Kenneth Davis and William Webb. A binomial coefficient congruence modulo prime powers. *J. Number Theory*, 43(1):20–23, 1993.

[5] Philip J. Davis. *Circulant Matrices.* Wiley, Chichester, 1979.

[6] Philip J. Davis. *Spirals : from Theodorus to chaos.* A.K. Peters, Wellesley, Mass., 1993.

[7] F. M. Dekking and M. Mendès France. Uniform distribution modulo one: a geometrical viewpoint. *J. Reine Angew. Math.*, 329:143–153, 1981.

[8] Peter Giblin and Matthew Trout. Symmetric and almost symmetric polygons. *The Mathematical Gazette*, 81(492):381–390, 1997.

[9] Pierre Goetgheluck. On prime divisors of binomial coefficients. *Math. Comp.*, 51(183):325–329, 1988.

[10] Branko Grunbaum and Geoffrey C. Shephard. *Tilings and patterns.* W.H. Freeman, New York, 1986.

[11] J. Konvalina and Y.-H. Liu. Arithmetic progression sums of binomial coefficients. *Appl. Math. Lett.*, 10(4):11–13, 1997.

[12] George E. Martin. *Transformation Geometry: An Introduction to Symmetry.* Springer-Verlag, New York, 1982.

[13] L. E. Mattics. Congruence mod p^k of sums of binomial coefficients. *Amer. Math. Monthly*, 93(9):740–741, 1986.

[14] Elmer G. Rees. *Notes on Geometry.* Springer-Verlag, Heidelberg, 1983.

[15] John Stillwell. *Elements of algebra: geometry, numbers, equations.* Springer-Verlag, New York, 1994.

[16] Hermann Weyl. *Symmetry.* Princeton University Press, New Jersey, 1962.